

Univerzita Karlova v Praze
Pedagogická fakulta

Bakalářská práce

Informační systém pro malotřídní školy
Information system for small schools

Adam Pátek

Katedra informačních technologií a technické výchovy

Vedoucí práce:	PhDr. Jiří Štípek, Ph.D.
Studijní program:	Specializace v pedagogice
Studijní obor:	Informační technologie se zaměřením na vzdělávání

Prohlašuji, že jsem bakalářskou práci na téma
Informační systém pro malotřídní školy vypracoval pod
vedením vedoucího bakalářské práce samostatně za použití
v práci uvedených pramenů a literatury. Dále prohlašuji, že
tato bakalářská práce nebyla využita k získání jiného nebo
stejného titulu.

V Praze 9. prosince 2016

.....

Podpis

Děkuji vedoucímu práce PhDr. Jiřímu Štípkovi,
Ph.D. za vstřícnost a především trpělivost při vedení
bakalářské práce.

ANOTACE

Předmětem bakalářské práce je vývoj softwarové aplikace pro sestavování rozvrhů místností, tříd a učitelů v prostředí málotřídních škol. Aplikace je realizována prostřednictvím programovacího jazyka Java SE s příslušnými knihovnamí a relačního databázového systému SQLite. Obsahuje analýzu nabídky software s podobným zaměřením, průzkum rozšíření těchto aplikací v prostředí českých málotřídních škol a přístupu ředitelů k jejich využívání společně se souhrnem základních požadavků a východisek funkcionality. Další část je podrobněji zaměřena na použité technologie, postup při vývoji back end části aplikace, tj. především návrh a řízení databáze a mechanismy pro kontrolu validity rozvrhu. Dále se zabývá vývojem uživatelského rozhraní aplikace a průvodce procesu inicializace uživatelských dat

KLÍČOVÁ SLOVA

informační systémy, malotřídní školy

ANNOTATION

The aim of this thesis is to develop a software application for timetable management in the small schools environment. The technologies used are Java SE programming language with various libraries and SQLite database system. The thesis contains an analysis of available software focused on timetabling, survey conducted among small schools in Czech Republic and resulting conclusions regarding the requirements for such application.

KEYWORDS

information system, small schools

Obsah

1	Úvod.....	6
2	Informační systémy ve školství.....	7
2.1	Informační systém.....	7
2.2	Školní informační systémy	7
3	Málotřídní školy	9
3.1	Definice a postavení málotřídních škol	9
3.2	Specifika málotřídních škol ve vztahu k IS	9
4	Srovnání existujících produktů.....	11
4.1	Bakaláři.....	12
4.2	aSc Rozvrhy	14
4.3	iŠkola	16
4.4	Škola online	17
4.5	Shrnutí srovnání IS	18
5	Výzkumná část	19
5.1	Cíl výzkumu.....	19
5.2	Hypotézy	19
5.3	Forma výzkumu	19
5.4	Administrace dotazníku	20
5.5	Zpracování dat	20
5.6	Interpretace výsledků	21
6	Východiska pro vývoj aplikace	26
6.1	Funkcionalita	26
6.2	User experience.....	27
6.3	Volba technologie	28
7	Vývoj.....	31
7.1	Uživatelské rozhraní	31
7.2	Databáze.....	36
7.3	Back end	37
7.4	Průběh a postup vývoje.....	42
7.5	Testování aplikace	42
8	Závěr.....	45
9	Přílohy	46
9.1	Příloha č. 1: Dotazník	46
10	Seznam použitých informačních zdrojů	48
11	Seznam příloh.....	51

1 Úvod

Příprava rozvrhu hodin je jen jednou z mnoha činností v širokém spektru administrativní zátěže ležící na vedení každé školy. Spolu s rozšiřováním informačních technologií ve školství vznikaly v průběhu času různé softwarové nástroje nabízející způsoby jak tento každoročně opakovaný proces usnadnit a urychlit. Když jsem ale příležitostně pomáhal s přípravou a správou webové prezentace jedné málotřídní školy ve východních Čechách, uvědomil jsem si, jak specifické požadavky tyto školy, v prostředí českého venkova rozšířené a tradiční, na takovou aplikaci mají.

Rozhodl jsem se, že svou zkušenost s málotřídkami využiji při zpracování bakalářské práce, ve které se pokusím zmapovat, do jaké míry se již software na tvorbu rozvrhů v málotřídkách rozšířil, jaké vlastně jsou představy ředitelů o funkcích a rozhraní těchto programů a na základě výsledků tohoto průzkumu společně s poznatky získanými srovnáním výše zmíněných rozšířených programů a jim podobných vytvořím autorský program, který bude praxi v málotřídních školách vyhovovat co možná nejlépe.

První kapitoly bakalářské práce jsou zaměřeny problematiku informačních systémů a jejich využití ve školství. Následně budou tyto informace zasazeny do kontextu málotřídních škol a popsána specifika požadavků málotřídních škol na informační systémy z pohledu vnějšího pozorovatele, včetně nastínění potenciálních problémů, které z těchto specifických požadavků mohou vznikat. Další část práce se věnuje nabídce školních informačních systémů s důrazem na analýzu toho, jak tyto systémy reflektují potřeby málotřídních škol. Pro získání kompletního obrazu o prostředí těchto vzdělávacích zařízení a jejich postavení ke školním informačním systémům především v kontextu tvorby rozvrhu je proveden výzkum mezi řediteli málotřídních škol s cílem potvrdit či vyvrátit hypotézy extrapolované z předchozích částí a také zjistit pohled ředitelů na věc.

Na základě teoretické a výzkumné části byly definovány požadavky na aplikaci vhodnou pro tvorbu rozvrhu v málotřídních školách, jejíž vývoj je předmětem zbytku práce.

2 Informační systémy ve školství

2.1 Informační systém

Informační systém (IS) je „ucelená skupina součástí pro sběr, ukládání a zpracování dat a pro poskytování informací, znalostí a digitální produktů“ (1). V kontextu této práce pak budeme mluvit o sadě softwarových nástrojů umožňujících ukládání a správu dat, včetně jejich následného zpracování a zpřístupnění, cílící na zjednodušení a zefektivnění řízení v prostředí organizací či institucí – u tohoto druhu informačních systémů se mluví o manažerských informačních systémech (MIS). Tyto nástroje obvykle představují nejrozličnější databázové aplikace, přičemž obsah databáze mohou tvořit např. informace o lidských zdrojích, skladech, účetní položky a další provozní či organizační data. Skutečnou podstatou informačního systému jsou potom relace mezi záznamy a možnosti jejich třídění, správy a extrakce. Pro skutečné využití informačního systému jsou nezbytné také nástroje pro export dat.

V informační společnosti tvoří MIS velmi důležitý nástroj rozhodovacího procesu v rámci managementu a setkáváme se s nimi jak v soukromých firmách, tak ve státní správě nebo ve školství, právě kterému se práce bude věnovat. MIS obvykle neslouží jen jako interní podpůrný nástroj pro vedoucí pracovníky, ale zajišťují efektivní kooperaci v rámci celé organizační struktury a zpravidla také umožňují generovat sestavy informací k veřejné prezentaci apod. Jednou z nejhodnotnějších funkcí MIS je pak schopnost zpracovávat shrnující zprávy poskytující ucelený, širší pohled na strategické informace (2).

2.2 Školní informační systémy

Školní informační systémy (ŠIS) jsou ve své podstatě zvláštním odvětvím manažerských informačních systémů a jako takové zajišťují podporu jak řízení školy jako takového, tak komunikace v rámci učitelského sboru, učitelů s žáky, ale i školy se zákonnými zástupci žáků a veřejností (3). Ve srovnání s podnikovými MIS jsou školní informační systémy navzdory své komplexnosti zpravidla nesrovnatelně levnější, a to zejména díky vysoké úrovni „standardizace“ těchto systémů. Zatímco firemní uživatelé často vyžadují modifikace dostupných systémů tak, aby vyhovovaly potřebám jejich konkrétního zaměření a specifikům jejich firemní struktury, školy mohou vzhledem k téměř shodným agendám a pevně dané struktuře využít univerzálně navržený produkt.

V průběhu vývoje ŠIS je signifikantní trend ve sjednocení aplikací, které dříve představovaly samostatné entity, do univerzálních, modulárních systémů – jeden IS tak může zahrnovat nejrůznější funkce, jako jsou evidence žáků, správa úvazků zaměstnanců, nástroje pro tvorbu rozvrhu, tisk vysvědčení a výkazů, vedení třídní knihy, správa knihovního fondu a podobně (4).

3 Málotřídní školy

3.1 Definice a postavení málotřídních škol

Málotřídní školy představují tradiční součást venkovského života na území České republiky a jako takové neplní jen funkci vzdělávacího zařízení. Škola v obci představuje velmi důležitý prvek občanské vybavenosti, venkovští učitelé historicky představovali v obci váženou osobu a měli, spolu např. s farářem, nezanedbatelný podíl na šíření vzdělání a kultury na venkově. (5)

V aktuální legislativě České republiky není pojem „málotřídní škola“ nijak definován, člení pouze školy s plným počtem devíti ročníků a ty, které všechny ročníky nemají (6). Vyhláška č. 48/2005 Sb. potom rozděluje školy tvořené jednou, dvěma, třemi a čtyřmi a více třídami prvního stupně a zároveň v § 5 umožňuje slučování tříd prvního stupně ZŠ: „Na prvním stupni školy lze do jedné třídy zařadit žáky z více než jednoho ročníku prvního stupně.“ (7)

Akademické a žurnalistické zdroje málotřídní školy definují právě jako školy, které mají méně tříd než ročníků, tj. kde jsou v jednotlivých třídách slučováni žáci z více ročníků (5) (8).

Z hlediska početnosti jsou málotřídky zcela jistě nezanedbatelnou součástí českého školství – představují přibližně 38% podíl (9) z celkových téměř 4000 základních škol (10) na našem území. Přes tuto svoji rozšířenost se ve školním roce 2004/2005 na málotřídkách vzdělávalo jen málo přes 10 % žáků (9).

3.2 Specifika málotřídních škol ve vztahu k IS

Ačkoli jsou z legislativního hlediska málotřídní školy vnímané prakticky stejně, jako školy plně organizované, a tudíž je na ně kladena srovnatelná administrativní zátěž a státní orgány od nich vyžadují stejné výstupy, v praxi jde o diametrálně odlišné instituce. Zatímco na úplných základních školách má ředitel často k ruce i několik pomocných či administrativních pracovníků a díky většímu počtu pedagogů může sám věnovat více času činnostem souvisejícím s organizací a managementem, ředitel školy málotřídní je ve výrazně složitějším postavení – běžně u něj učení představuje přes 40 % úvazku a přitom je na většinu činností spojených s řízením školy sám (11, s. 148-149). Tato situace přitom není výjimečná pro ředitele českých málotřídek, v průzkumu probíhajícím v devadesátých letech mezi řediteli venkovských škol ve Velké Británii se objevují podobné závěry (12).

Ve světle těchto skutečností lze předpokládat, že energie ředitelů bude věnována spíše otázkám přímo ovlivňujícím chod školy nebo vyžadovaným legislativou či nadřízenými orgány, než procesním záležitostem. Jako ideální se proto jeví takový informační systém, který vyžaduje co nejmenší rozsah počáteční konfigurace, zároveň je uživatelsky přívětivý s co nejstrmější křivkou učení a s ohledem na to, že pro ředitele, který jej používá, představuje management jen něco málo přes půl úvazku. Dále je třeba brát v úvahu skutečnost, že ačkoli je obsah administrativy na málotřídce srovnatelný s velkými školami, soubory dat, které musí informační systém zpracovat, jsou řádově menší. Ve výsledku se tudíž může ukázat, že příliš „chytré“ systémy s množstvím funkcí mohou svou složitostí a náročností nastavení a nasazení práci ředitele spíše zkomplikovat než ulehčit.

4 Srovnání existujících produktů

Během srovnání programů běžně dostupných na trhu se zaměřím především na dvě kritéria zásadní pro málotřídní školy – uživatelskou přívětivost a funkce potřebné pro zanesení jevů pro málotřídky specifických do rozvrhu, tj. především spojování tříd.

Pro zjištění a demonstraci těchto vlastností se s použitím nápovědy k jednotlivým programům pokusím replikovat mírně zjednodušený (vynechání dobrovolných kroužků, apod.) rozvrh jedné skutečné málotřídky.

		1.	2.	3.	4.	5.	6.	7.
Pondělí	1.	Čj	M	Čj	Tv			
	2.	Čj	M	Čj	Tv			
	3.	Čj	M	Čj	Prv			
	4.	Čj	M	Čj	Aj	VI		
	5.	Čj	M	Čj	Aj	VI		
Úterý	1.	M	Čj	Prv	Pv		Hv	
	2.	M	Čj	Prv	Pv		Hv	
	3.	M	Čj	Prv	Tv		Hv	Čj
	4.	M	Čj	Př	Tv		Hv	
	5.	M	Čj	Př	Tv		Hv	Čj
Středa	1.	Tv	Čj	Čj				
	2.	Tv	Čj	Čj				
	3.	Aj	Čj	M	Čj	Pv		
	4.	Aj	Čj	M	VI	Pv		
	5.	Aj	Čj	M	VI	Pv		
Čtvrtek	1.	M	Čj	ŠD	Čj		Vv	Tv
	2.	M	Čj	ŠD	Čj		Vv	Tv
	3.	M	Čj	Aj	Vv		Aj	Čj
	4.	M	Čj	Aj	Vv		Vv	Čj
	5.	M	Čj	Aj	Vv		Vv	Inf
Pátek	1.	Čj	M	Prv	Čj			
	2.	Čj	M	Prv	Čj			
	3.	Čj	M	Prv	Tv			
	4.	Čj	M	Př	Tv			
	5.	Čj	M	Př	Tv			

Obr. 1 - Referenční rozvrh

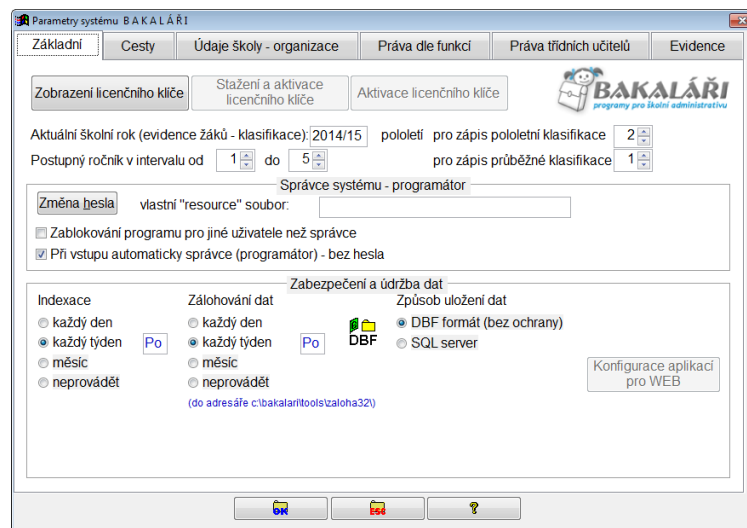
Vzhledem k obousměrné závislosti získaných informací (výsledky dotazníku ovlivnily výběr studovaných programů a zároveň bylo pro sestavení dotazníku nezbytné mít přehled o funkcích a vlastnostech existujících systémů) probíhala tvorba této kapitoly simultánně s dotazníkovým šetřením. Z toho vyplývá, že jsou zde uvedeny některé softwarové nástroje, které jsou rozšířeny mimo prostředí málotřídek, ale v datech z dotazníku se neobjevují. V pozdější fázi jsem se zároveň snažil i mezi méně rozšířenými produkty najít takový, který by vůbec sestavení rozvrhu, jaký organizace vyučování v málotřídních školách vyžaduje, umožňoval.

4.1 Bakaláři

„Bakaláři“ je název univerzálního modulárního software poskytujícího nástroje pro kompletní organizační administrativu ve školách všech velikostí, od základních škol až po vyšší odborné. Mezi jeho funkce kromě generátoru rozvrhů patří také evidence žáků a zaměstnanců, školní matrika, webová žákovská knížka, rozvrh hodin, suplování, plán školních akcí, rozpis maturit, třídní kniha, tematické plány, přijímací zkoušky, knihovna, inventarizace a poté další uživatelské doplňky. Žáci a rodiče také mají možnost využít zdarma mobilní aplikaci pro zařízení s operačním systémem Android, která nabídne pohled do žákovské knížky, rozvrhu hodin a dalších agend. První zakoupení společného prostředí a rozvrhového modulu vyjde k 1. listopadu 2014 základní školu s méně než sto žáky na 4020 Kč včetně manipulačního poplatku, roční upgrade potom 900–1290 Kč.

Ačkoli jsou Bakaláři programem velmi všestranným a zpracovaným, trpí na velmi komplikované a poměrně nepřehledné uživatelské rozhraní. Tento jev se bohužel projevuje i při používání samotného rozvrhového modulu, protože jeho běh vyžaduje společné prostředí navržené pro práci s kompletní instalací. Navíc software postrádá přehledného průvodce inicializací vstupních dat, což ústí ve velmi pozvolnou křivku učení, obzvlášť pokud se uživatel snaží zákonitosti programu pochopit bez vedení ze strany instruktora.

První, co se po spuštění čisté instalace zobrazí, je dialogové okno s nastavením parametrů programu, které obsahuje jak organizační údaje týkající se přímo školy, tak čistě systémová nastavení jádra Bakalářů, přičemž všechny tyto možnosti postrádají jakékoli indície, zda jsou pro běh jednotlivých modulů nezbytné, nebo mohou zůstat nevyplněné či ve výchozím stavu. Když jsou všechny hodnoty nastaveny, je možné pokračovat do společného prostředí modulů Rozvrh a Evidence. Zde ještě uživatel nenalezne rozhraní k samotné tvorbě rozvrhu – tato část představuje prakticky prázdné okno s nástrojovými panely. Pod odpovídajícími nabídkami (Datové soubory → Učitelé, Třídy, Místnosti apod.) lze nastavit seznamy jednotlivých entit zapojovaných do rozvrhových lístků, zároveň jsou zde nástroje k přípravě studijních plánů a rozdělení úvazků učitelů, na základě kterých se samotné rozvrhové lístky generují.



Obr. 2 - Bakaláři: parametry systému

Přestože jsou jednotlivá okna pro zadání parametrů těchto entit opět poměrně komplexní, při vytváření rozvrhu není nutné vyplňovat vše. Stačí zadat pouze základní informace, jako je jméno u učitelů, název u předmětů apod. Na okně třídy je možné například demonstrovat roztržistěnost jednotlivých ovládacích prvků mezi moduly, protože ačkoli se zde nachází zaškrtačací pole „spojena“ umožňující nastavit spojení jednotlivých tříd typické právě pro málotřídky, skutečné rozvrhové spojení se provádí až v rámci přidělování úvazků a to na úrovni jednotlivých rozvrhových lístků. Tímto způsobem lze navíc sloučit jen rozvrhové lístky tříd se stejnou hodinovou dotací spojovaného předmětu, takže se mi ani s pomocí dodávaného manuálu nepodařilo přesně replikovat referenční rozvrh. Požadovaného výsledku by pravděpodobně šlo dosáhnout např. pomocí duplikování učitelů nebo využitím práce se skupinami v rámci tříd, nicméně takový postup už by byl poměrně komplikovaný, v případě duplicitních učitelů navíc nekorektní z hlediska zachování konzistence dat.

Třída	Aj	CJ	Hv	Inf	M	Prv	Pv	Vi	Tv	Pr	Vv	Hod. přiděl.
1.	KR	JN			KR	KR	KR		MŠ		MŠ	21+0 (21+0)
2.	KR	JN			KR	KR	KR		MŠ		MŠ	21+0 (21+0)
3.	KR	JN	JN		JN	KR	KR		MŠ		MŠ	25+0 (24+0)
4.	KR	JN	JN		JN		KR	JN	MŠ		MŠ	25+0 (25+0)
5.	KR	JN	JN		KR	JN	MŠ		MŠ		MŠ	26+0 (26+0)
XXXX	9+0	23+9	3+1	1+0	15+4	7+0	3+1	4+0	12+0	4+0	7+0	XXXXX

Obr. 3 – Bakaláři: nastavení úvazků (zobrazení třída x předměty)

V průběhu zpracovávání bakalářské práce byla vydána nová verze systému Bakaláři, která má přinést modernizaci uživatelského rozhraní a zpříjemnění práce. Přestože však na první pohled rozhraní vypadá o něco přístupněji než původní, reálně vyžaduje stejná vstupní data a i ovládání zůstalo vesměs stejné. Funkcionalita zůstala taktéž nezměněna, takže je z hlediska použitelnosti pro málotřídní školy podrobný rozbor tohoto upgradu irelevantní.

4.2 aSc Rozvrhy

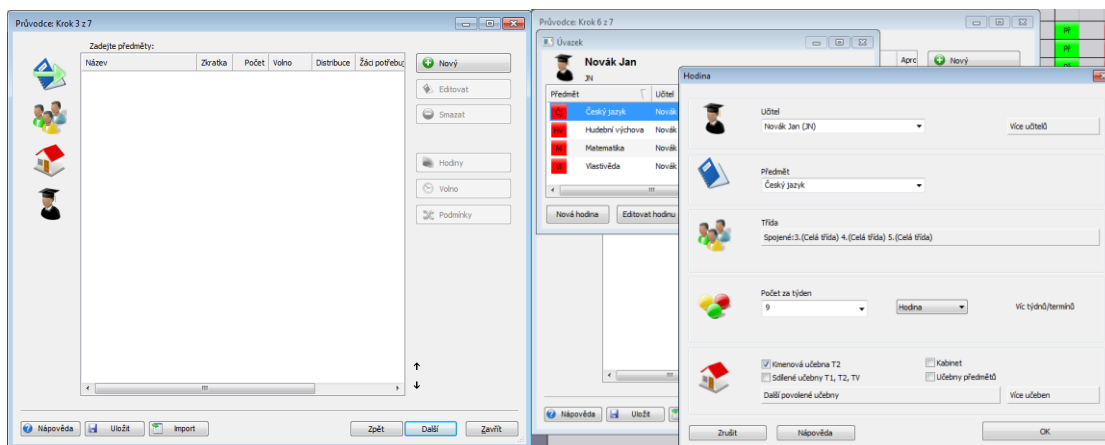
Na rozdíl od Bakalářů se aSc Rozvrhy profilují jako úzce zaměřený nástroj určený čistě k vytváření rozvrhu a souvisejícím činnostem. V této oblasti autoři programu disponují dvacetiletou zkušeností a nabízí také podporu specifických požadavků různých zemí. Stejně jako u Bakalářů, lze i v případě aSc Rozvrhů k prohlížení rozvrhu použít mobilní zařízení. Cena licence aSc Rozvrhy pro základní školu je 4400 Kč.

Vzhledem k jasnému určení softwaru je již první použití výrazně intuitivnější, než je tomu u předchozího testovaného programu. Hned na hlavní obrazovce zobrazené po spuštění je několik výrazných a srozumitelných ikon nabízejících přístup jak přímo k započetí přípravy rozvrhu, tak k podrobnému online návodu, ukázkovým rozvrhům a dalším informacím týkajícím se programu a jeho tvůrců. Navíc je zde možnost spustit úvodní lekci používání programu, která probíhá přímo v uživatelském prostředí. Zde virtuální průvodce Josef Školský novému uživateli velmi srozumitelnou cestou vysvětlí, k čemu slouží jednotlivé funkce a jak s tvorbou rozvrhu začít. Reálně je možné tuto ukázkou přeskóčit, celá tvorba rozvrhu totiž probíhá prostřednictvím uživatelsky přívětivého průvodce.



Obr. 4 – aSc Rozvrhy: úvodní obrazovka

Během procházení tohoto průvodce jsou zadány všechny údaje nezbytné k vygenerování rozvrhových lístků, tj. základní údaje o škole a rozvrhu (školní rok, počet hodin apod.), seznam tříd, učeben, předmětů, a učitelů. Během přidávání učitelů lze také nastavit jednotlivé úvazky včetně spojení tříd. Tento přístup se na pohled sice jeví velmi výhodně a ve velkých školách zřejmě představuje pozitivum, v málotřídkách, kde musí být brán ohled na další rozměr rozvržení – spojování tříd – však představuje určitou komplikaci, protože ztěžuje dynamické přizpůsobení během skutečného rozvrhování. Funkce spojení jako taková navíc opět neumožňuje spojit různé předměty, resp. třídy s různou hodinovou dotací daného předmětu, do jednoho rozvrhového lístku. Ve výsledném rozvrhu se tohoto efektu dá sice dosáhnout vytvořením duplicitních záznamů a vynucením rozvrhové kolize, kterou program vyhodnotí, nicméně tento postup přípravu rozvrhu a ovládání celého softwaru komplikuje, navíc dává prostor k chybám.



Obr. 5 – aSc rozvrhy: průvodce tvorbou rozvrhu

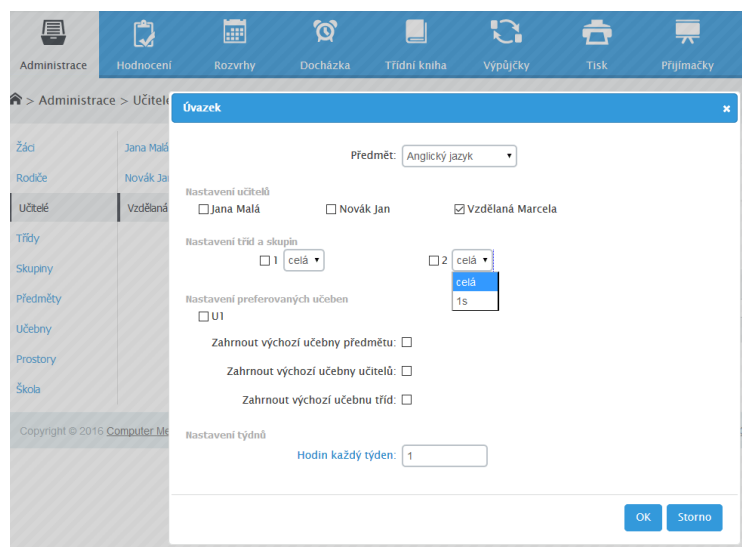
4.3 iŠkola

Produktem výrazně odlišným od obou předchozích je iŠkola.cz. Na rozdíl od nich jde totiž o webovou aplikaci přístupnou odkudkoli prostřednictvím webového prohlížeče. Díky tomu má několik zásadních výhod. V první řadě zbavuje školu požadavků na výpočetní zdroje, stejně tak jako jejich správu a údržbu. Všechna data školy jsou díky tomu uložena na serverech poskytovatele a aplikace je poskytována jako služba. Webové prostředí s sebou zároveň přináší nezávislost na platformě (včetně možnosti přístupu a správy z mobilních zařízení, tabletů apod.) a přístupnost odkudkoli, kde je dostupný internet. Cena služby pro školu do 50 žáků je 1200 Kč ročně včetně DPH (13).

Informační systém iŠkola podobně jako Bakaláři poskytuje kompletní školní agendu, mj. žákovskou knížku, třídní knihu, školní matriku, vysvědčení apod. Zároveň je možné nastavit přihlašovací údaje pro rodiče a žáky, kteří pak mohou prohlížet známky, suplování apod. Přehledné a moderní rozhraní, k jehož procházení slouží hlavní horizontální menu doplněné vertikálním submenu pro každou kategorii, umožňuje velmi rychlou orientaci v systému. Dokumentaci tvoří sada příruček ve formátu PDF, které jsou dostupné z webu a i přes svůj rozsah (několik set stránek) jsou velmi přehledné.

Jedním z dostupných modulů je i rozvrh, který zde neobsahuje funkci automatického generování. Stejně jako v předchozích dvou programech pracuje rozvrhový modul na principu vytvoření rozvrhových lístků generovaných z úvazků učitelů a nesoucích veškeré informace o hodině – vyučovanou třídu, místnost, kde výuka probíhá, vyučujícího a předmět. Ty už se pak jen nasazují do konkrétních dnů a časů.

Dialog pro tvorbu úvazku, ke kterému je možné dostat se několika způsoby (z konfigurace učitele, třídy, skupiny) nabízí rozdělení tříd do skupin (ty musí být nakonfigurovány předem), sloučení tříd i sloučení skupin z různých tříd. Lze také přiřadit jedné hodině dva učitele (např. asistent pedagoga). Bohužel zde znovu chybí možnost sloučit dva předměty v rámci jedné hodiny, nicméně je na tuto možnost pamatováno v dokumentaci. Navržené řešení sestává z vytvoření nového předmětu, který je pojmenován např. „Prvouka+Přírodověda“ a ten je potom nasazen do rozvrhu. (14, s. 60-63). Toto řešení je sice funkční a šlo by aplikovat i na předchozí aplikace, nicméně jednoznačně není optimální ani intuitivní.



Obr. 5 - iŠkola: nastavení úvazku

Další variantou, jak v rozvrhu docílit „spojení“ hodin, je nastavit pro každý ze spojených předmětů odpovídající úvazky, ty nasadit do rozvrhu „přes sebe“ a zanedbat hlášené konflikty (tj. v jeden čas v jedné učebně učí učitel několik různých hodin). Tento postup ovšem potom v systému vytvoří každou z učených „částí“ zvláštní úvazkové hodiny, tj. ve výsledku by se úvazek učitele jevil jako několikanásobný ve srovnání se skutečným stavem.

4.4 Škola online

Škola online je dalším zástupcem webových informačních systémů poskytovaných ve formě služby s kompletním technologickým zajištěním na straně poskytovatele. Balíček obsahující pouze jádro systému, školní matriku, evidenci osob a rozvrhový modul (včetně suplování) lze pro školu do 100 žáků získat za roční poplatek 2600 Kč vč. DPH, kompletní sada funkcí je potom za 4600 Kč/rok (15).

Jde opět o velmi komplexní a univerzální systém poskytující jednotné řešení pro celou školu včetně nástrojů pro komunikaci mezi školou, rodiči a žáky. Filozofií ovládání se do značné míry podobá předchozí aplikaci, což je vzhledem ke stejnému funkčnímu modelu přirozené a pochopitelné.

Rutina související s tvorbou rozvrhu je taktéž velmi podobná předchozím zmíněným, sestává z přípravy učitelů, tříd, předmětů, volitelně též učeben, skupin atp., ze kterých se poté vytvoří úvazky vázané na učitele a ty se následně nasadí do rozvrhu. Tento princip bohužel opět implikuje značné problémy při spojování tříd – i zde je možnost přiřadit jednomu rozvrhovému lístku více tříd, ale ani zde není možné přiřadit více předmětů. Opět

lze tuto systémovou limitaci různými způsoby obejít, nicméně narazíme na stejné potíže, jako u předchozích.

Úvazky učitelů

Učitel: Malá Jana - MJ ✖ Novák Jan - nj

Předmět: ČJ-Český jazyk

Třída: 1 - celá třída ✖
2 - celá třída ✖

Učebna: Pro přiřazení učebny k úvazku stiskněte tlačítko [<] na pravé straně.

Počet za týden: 4 Délka: 1 hodina

Cyklus: bez cyklů

Platná celé období: ☒

Uložit Uložit a nový Uložit a nový podobný Zpět

Obr. 6 - Škola online: nastavení úvazku

4.5 Shrnutí srovnání IS

Ze srovnání čtyř prominentních softwarových produktů poskytujících funkcionalitu nezbytnou k sestavení rozvrhu je evidentní, že se při vývoji tohoto druhu aplikací berou v úvahu především potřeby velkých, resp. plně organizovaných základních, středních a vyšších odborných, škol. To z nich nepochybně dělá neocenitelné nástroje pro velkou část školských zařízení, ale prostředí málotřídek je do velké míry opomíjeno.

Ve finále by se u většiny z nich sestavení odpovídajícího rozvrhu podařilo, způsoby, jakými toho lze docílit, jsou ovšem nestandardní a značně komplikované. Zároveň jdou více či méně proti smyslu těchto nástrojů a potlačují funkci mechanismů, které mají v této činnosti pomáhat – mj. je vyřazována kontrola rozvrhových kolizí, znehodnocují se sestavy úvazků učitelů. Navíc by jejich pořízení a údržba pro školy představovaly nezanedbatelné náklady jak finanční, tak časové a personální.

V odpovědích respondentů výzkumu v kapitole 5.5 Zpracování dat měl poměrně výrazné zastoupení také systém společnosti DM Software, který by bylo na místě také zařadit do srovnání, nicméně při pokusu společnost DM Software kontaktovat jsem nedostal žádnou odpověď. Je ale očekávatelné, že kdyby měl tento IS intuitivní a přehledný rozvrhový modul, byl by ve školách, kde je používán k evidenci, také nasazen.

5 Výzkumná část

5.1 Cíl výzkumu

Pro navržení aplikace, která by co nejlépe vystihovala potřeby málotřídních škol, bylo nezbytné ověřit a upřesnit předpoklady vycházející ze zkušeností, které jsem získal setkáváním se s prostředím málotřídek v minulosti. Zároveň se jevilo žádoucím zajistit si alespoň základní vhled do stavu na trhu s podobnými aplikacemi, tj. dovědět se, zda vůbec existuje prostor pro aplikaci zaměřenou na tvorbu rozvrhů v málotřídkách a poptávka po takové aplikaci.

Za vedlejší přínos výzkumu lze potom považovat zvýšení povědomí ředitelů o existenci daného druhu aplikací a poskytnutí možnosti vyzkoušet výsledný produkt pro vlastní školu.

5.2 Hypotézy

K výzkumné části jsem přistupoval s následujícími předpoklady, jejichž potvrzení či vyvrácení mělo mít zásadní vliv nejen na vlastnosti budoucí aplikace, ale mohlo ukázat, že takový produkt ve skutečnosti vůbec nemá ve školách místo:

1. V málotřídních školách převažuje manuální tvorba rozvrhů, ať už v papírové formě nebo např. v tabulkových procesorech, a to z těchto důvodů:
 - a. Dostupné nástroje jsou neúměrně komplikované vzhledem k potřebám málotřídních škol
 - b. Finanční prostředky škol jsou preferenčně směřovány jinam, než do nákupu nástrojů používaných maximálně několikrát ročně
2. Stávající aplikace jsou zaměřené především na velké školy, a tudíž nemusí obsahovat funkce reflektující požadavky těch málotřídních

5.3 Forma výzkumu

Jako nejvhodnější řešení jsem vyhodnotil provedení dotazníkového šetření přímo u ředitelů málotřídek, které poskytuje snadno kvantifikovatelná a jednoznačná data (16, s. 54), ale zároveň zachovává určitou otevřenost otázek žádoucích v tak diverzifikovaném prostředí, jaké málotřídní školy představují. Online dotazník v takovém rozsahu, jaký byl zvolen, navíc od zaneprázdněných ředitelů vyžaduje jen několik málo minut času, což mělo přispět k vyšší návratnosti odpovědí.

Šetření bylo provedeno prostřednictvím polostrukturovaného interaktivního dotazníku postaveného na platformě Google Forms. Ředitelé, kteří se rozhodli participovat na průzkumu, byli postaveni před pět otázek, které se ale postupně rozcházely v závislosti na jednotlivých odpovědích. Všechny otázky byly uzavřené, některé z nich umožňující pouze jednu odpověď (právě ty mj. umožnily personalizaci následujících otázek), některé více různých odpovědí a některé obsahovaly i možnost „jiné“ odemykající textové pole pro vlastní tvořenou odpověď. Pokud se na konci dotazníku respondent rozhodl projevit zájem o vyzkoušení finální aplikace, měl možnost přiložit kontaktní e-mailovou adresu (celý dotazník viz příloha 10.1 Dotazník).

5.4 Administrace dotazníku

Oslovení ředitelů jsem realizoval prostřednictvím e-mailu rozeslaného do 115 málotřídních škol z celé České republiky náhodně vybraných ze seznamu více než 1400 málotřídek získaného prostřednictvím Odboru školské statistiky, analýz a informační strategie Ministerstva školství, mládeže a tělovýchovy ČR. Náhodná volba oslovených zajistila stejnou pravděpodobnost účasti pro málotřídní školy na území celé republiky (16, s. 53) s cílem zabránit např. takové situaci, kdy by si několik vzájemně blízkých ředitelů – respondentů navzájem řeklo o určité aplikaci, přičemž by se pak její zastoupení v šetření jevilo jako signifikantní, ačkoli by reálně bylo v celkovém měřítku minimální.

Ze 115 rozeslaných byl jeden e-mail vrácen poštovním systémem z důvodu neplatné e-mailové adresy. Při uzavření výsledků registroval dotazník celkově 43 odpovědí, což představuje více než 37% návratnost.

5.5 Zpracování dat

Vzhledem k možnosti tvořených odpovědí bylo třeba tyto zanalyzovat, vyhodnotit, zda ve skutečnosti nejsou ekvivalentní těm předpřipraveným, a zhodnotit jejich relevanci v kontextu ostatních odpovědí a tématu práce. Takové odpovědi, které se objevily opakovaně, byly sjednoceny a vyčleněny jako další jednotná skupina tak, aby nezařaditelné a nezpracovatelné „ostatní“ odpovědi představovaly co možná nejmenší část. Dále bylo nezbytné vyřadit invalidní odpovědi, mj. takové, kde respondent nesprávně pochopil otázku.

Pro usnadnění interpretací potom jsou výsledky zpracovány do několika tabulek podle potřeby doplněných grafy.

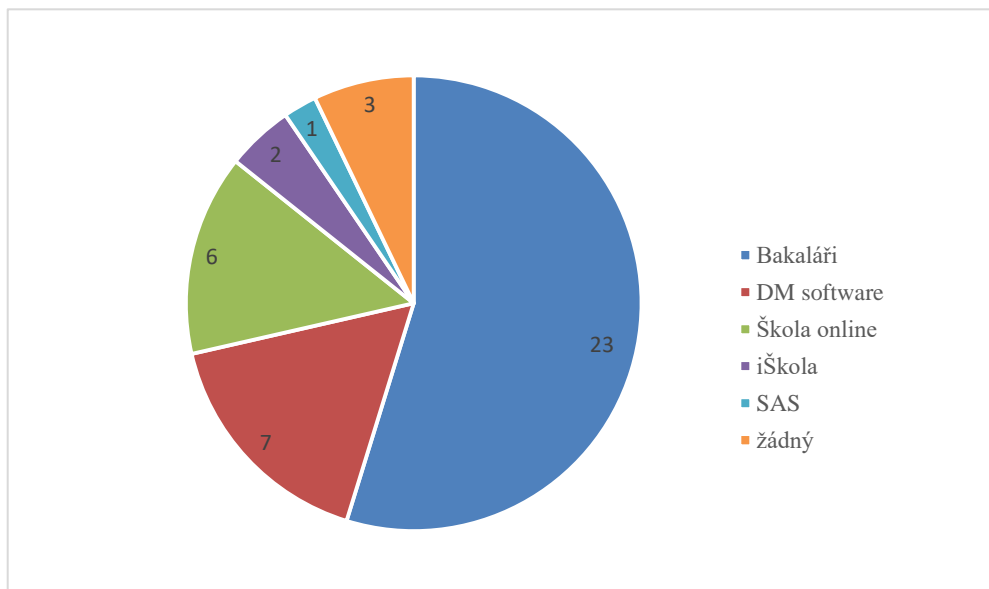
5.6 Interpretace výsledků

Otázka 1: Jaký program pro zpracování informací ve škole používáte?

První otázka mírně vybočuje ze zaměření práce, jejím účelem bylo primárně zasadit zbylá data do širšího kontextu a zhodnotit penetraci prostředí málotřídek informačními systémy v širším smyslu. Zároveň posloužila jako vodítko při volbě aplikací testovaných v kapitole 4 Srovnání existujících produktů.

Tabulka 1 - Zastoupení IS v málotřídních školách

Informační systém	Absolutní četnost	Relativní četnost [%]
Bakaláři	23	54,76
DM software	7	16,67
Škola online	6	14,29
Žádný	3	7,14
iŠkola	2	4,76
SAS	1	2,38
Suma	42	



Graf 1 - Zastoupení IS v málotřídních školách

Ze získaných dat je zřejmé, že informační systémy jako takové jsou na málotřídkách využívány v naprosté většině případů, „žádný“ v nějaké formě odpovědělo 7,5 % respondentů, přičemž jednou ze získaných odpovědí bylo „Klasické papírové rozvrhy“, což

zřejmě značí nesprávné pochopení otázky a nevylučuje stejný jev u ostatních tří. Tuto evidentně neplatnou odpověď jsem ze zpracování výsledků vyřadil.

*Otázka 2: Které z následujících prostředků specifických pro
malotřídky používáte ve svých rozvrzích?*

Zaměření druhé otázky se přímo dotýká funkcionality nezbytné pro přípravu rozvrhu na malotřídní škole. Odpovědi k ní přiřazené se spíše doplňují, než aby nezbytně popisovaly dvě různé varianty, přičemž obě měly za úkol ověřit problematičnost využití stávajících aplikací v ekosystému malotřídních škol – výsledek ukazuje, že ze 43 zúčastněných škol jsou dostupné varianty schopny bez kompromisů a komplikací uspokojit potřeby pouze v jediném případě, zatímco u 27 z nich by efektivní nebylo ani improvizované řešení navrhované v dokumentaci iŠkola.cz (viz kapitola 4.3 iŠkola).

Tabulka 2 - Prostředky používané v rozvrzích malotřídek

Prostředky používané v rozvrzích malotřídek	Absolutní četnost	Relativní četnost [%]
Spojování různých předmětů v rámci jedné vyučovací hodiny	15	34,88
Spojování předmětů s rozdílnou hodinovou dotací mezi spojenými ročníky	10	23,26
Obojí	17	39,53
Ani jedno z uvedených	1	2,33

*Otázka 3: Používá Vaše škola specializovaný počítačový program
pro sestavování rozvrhu?*

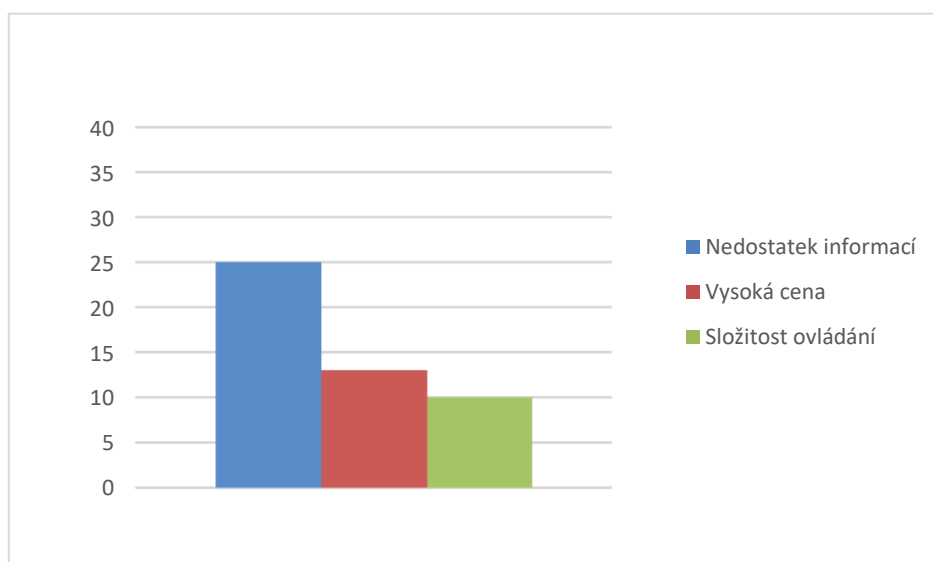
Tato otázka byla místem, kde se měli respondenti rozdělit na dvě skupiny – ty, kteří rozvrhy připravují s pomocí dostupných softwarových nástrojů a ty, kteří je tvoří tradičně na papír, případně prostřednictvím nějakých univerzálních nástrojů typu tabulkového procesoru. Ukázalo se však, že, minimálně v rámci zkoumaného vzorku malotřídních škol, tato dualita ve skutečnosti neexistuje, a všech 43 získaných odpovědí bylo „Ne“. Tento výsledek poměrně jasně vypovídá o potenciálu uplatnění aplikace.

Otázka 4: Označte, prosím, důvody, které vás odrážejí

Čtvrtým prvkem dotazníku byla otázka umožňující označit několik odpovědí zároveň – tři předpřipravené na základě vlastních hypotéz či předpokladů a čtvrtou ve formě textového pole pro neočekávané varianty. Cílem této otázky bylo zjistit, jak co nejlépe vystihnout problémy, které brání využití aplikací v málotřídních školách, při samotném vývoji.

Tabulka 3 - Překážky v užívání specializovaných programů k přípravě rozvrhů

Překážky v užívání specializovaných programů k přípravě rozvrhů	Absolutní četnost	Relativní četnost [%]
Nedostatek informací o nabídce aplikací tohoto druhu	25	58,1
Příliš vysoká cena v poměru s reálným přínosem takové aplikace	13	30,2
Složitě ovládání nabízených produktů	10	23,3
Ostatní	7	16,3



Graf 2 - Překážky v užívání specializovaných programů k přípravě rozvrhů (absolutní četnost)

Zde lze za překvapivý považovat pocit nedostatečné informovanosti mezi respondenty, a to především vzhledem k výstupu z první otázky, kde výrazná většina z nich uvedla informační systém, který v nějaké formě nástroj pro tvorbu rozvrhů obsahuje. Tento výsledek nicméně nevylučuje platnost žádné z hypotéz.

Druhé dvě odpovědi cílí přímo na ověření platnosti předpokladů, a ačkoli jejich zastoupení není zcela přesvědčivé, není ani zanedbatelné. V každém případě představuje potenciální prostor pro uplatnění práce.

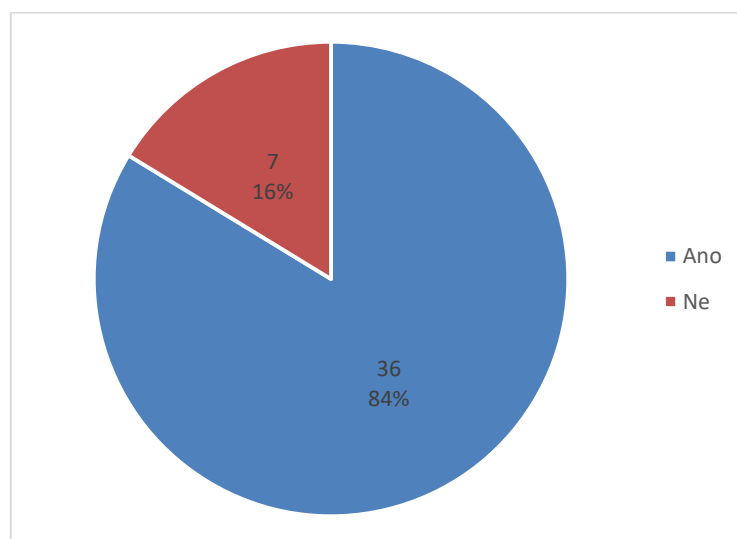
Zásadní roli v této otázce má i možnost „ostatní“, jejímž cílem bylo umožnit získání informací, které nejsou v předpokladech vůbec zahrnuty a potenciálně by mohly přinést zcela nový pohled do problematiky. Výsledné odpovědi byly následující (neupravené, autentické texty):

- „Zatím nebylo nutné, Excel postačil...“
- „baví mě to“
- „Nevidím v tom žádný přínos, rozvrh si dokážu sestavit sama.“
- „Zvládáme poměrně snadno pomocí tužky a papíru:-)“
- „máme vytvořený svůj vlastní "manuální systém", který nám vyhovuje“
- „Sestavování rozvrhu bylo zkušeno v programu Bakalář, ale ten nedovedl sestavit rozvrh pro sloučené ročníky“
- „Je to tak ,zaběhnuté““

V obecném pohledu opět ukazují jistou neochotu, resp. absenci potřeby, ředitelů málotřídek investovat (jak materiálně, tak personálně) do komplikovaných řešení a stejnou zkušenost s jejich nevhodností, která je prezentována v kapitole 4 Srovnání existujících produktů této práce.

Otázka 5: Máte zájem o vyzkoušení volně dostupné aplikace pro tvorbu rozvrhů zaměřené přímo na málotřídní školy?

V poslední části dotazníku jsem se rozhodl již přímo zmapovat případné zájemce o svou práci a její potenciální přínos. Výsledky ukazuje následující graf:



Graf 3 - Zájem o vyzkoušení aplikace

Poměrně zajímavý je zde kontrast mezi neinformovaností ředitelů a zájmem o další informace. Zde mne napadají dvě situace, které jej mohou vysvětlovat. Buď jsou se svým aktuálním řešením spokojeni a necítí potřebu jej měnit, ale v principu se nebrání návrhům

na inovace (na tuto variantu poukazují i odpovědi z předchozí otázky), nebo sice aktuální proces vnímají jako suboptimální, ale nepředstavuje pro ně takový problém, aby aktivně hledali alternativu. V každém případě výsledky této sekce dotazníku představují značný potenciál pro využití aplikace vznikající v rámci této práce a získané kontaktní e-mailové adresy představují nezanedbatelnou počáteční uživatelskou základnu.

6 Východiska pro vývoj aplikace

6.1 Funkcionalita

Jak je uvedeno již v kapitole 2.2 Školní informační systémy, dlouhodobým trendem je slučování jednotlivých aplikací pro podporu manažerských procesů do modulárních, propojených informačních systémů. V raných fázích plánování práce jsem proto i já počítal s implementací funkcí umožňujících alespoň import dat přinejmenším z těch nejčastěji zastoupených školních informačních systémů na trhu, a pokud by se ukázal jako žádoucí, také export do nějakého kompatibilního formátu. Stejně tak jsem uvažoval nad rozsahem automatizace přípravy rozvrhu – jak vyřešit plánování přestávek, rozdělení různých předmětů v rámci dne, hodnocení vhodnosti návaznosti jednotlivých předmětů na sebe. Během srovnávání aplikací a vyhodnocení výsledků dotazníkového šetření se ale ukázalo, že tyto funkce jsou v prostředí malotřídek nejen zbytné, ale navíc zhoršují uživatelskou přívětivost softwarových produktů a mohou být kontraproduktivní i z časového hlediska, kdy zadání jednotek tříd a učitelů může být rychlejší, než export ze stávajícího informačního systému a nový import do rozvrhové aplikace.

Naopak nezbytnou součástí je možnost nasadit více předmětů a více tříd s jedním vyučujícím do jedné vyučovací hodiny. Jako nejvhodnější řešení se jeví vnímat jednotlivé části sloučené třídy jako diskrétní hodiny. Úvazky se nebudou tvořit předem, ale až ad hoc při nasazování do konkrétních časových plánů, kdy bude předmět, vyučující a učebna tvořit nezávislé entity, jejich relace se bude nastavovat jen pro konkrétní hodinu.

Dalším prvkem bude kontrolní mechanismus rozvrhu, který zde bude velmi jednoduchý, protože jedinou věcí, kterou lze jednoznačně z rozvrhu málotřídní školy vyloučit je přítomnost jednoho učitele ve více učebnách zároveň. Ostatní potenciální kolize, jako například dvě vyučovací hodiny ve stejný čas, jedna třída ve dvou učebnách a podobně jsou vyloučeny již samotnou filozofií a designem aplikace.

Přístup k počítání úvazků pro případné administrativní sestavy pak bude čistě deskriptivní – aplikace bude obsahovat funkci pro reporting úvazků učitelů a vytíženost tříd. Soupis úvazků potom bude uzpůsoben tak, aby paralelně probíhající výuku jednoho učitele (tj. např. hodinu spojenou ze tří tříd a dvou předmětů) počítal korektně podle skutečného odučeného času, ne podle počtu předmětů a tříd. Tato funkcionalita umožní především snadnou kontrolu naplnění úvazků učitelů a zároveň plnění školního vzdělávacího programu v rozvrzích jednotlivých tříd.

Ve výsledku se tudíž nebude jednat o plnohodnotný informační systém, ale spíš jednoúčelovou aplikaci, funkčností odpovídající jednomu modulu běžného IS, doplněnou o funkce pro vytěžení informací a jiné podpůrné funkce.

Z hlediska možných výstupů potom aplikace bude obsahovat export rozvrhů do HTML souboru, včetně FTP klienta pro přímý upload rozvrhů na web školy. Pro tisk rozvrhů je zde druhá možnost exportu do formátu PDF. Výsledným přínosem tudíž nebude jen zjednodušení samotného rozvržení výuky, proti stávajícím způsobům rozšířeným mezi málotřídkami, ale také efektivní možnost zveřejnění a zpracování výsledných dat – touto částí se program do jisté míry mezi informační systémy zařadí.

6.2 User experience

Uživatelská přívětivost je v tomto projektu jedním z nejdůležitějších kritérií, bezmála tak důležitým, jako je samotná funkční podpora specifik málotřídních škol. V návrhu rozhraní a ovládání aplikace jsem proto vyšel ze zkušeností získaných testováním komerčně dostupných aplikací navržených plně profesionálními vývojáři, ale zároveň jsem se snažil, aby výsledný produkt co nejvíce reflektoval pracovní postup přípravy rozvrhu na málotřídních školách podle obrazu, který jsem si udělal z odpovědí na dotazníkové šetření.

Jako centrální prostředek k docílení co nejmenší časové investice do prvotní inicializace databáze a k rychlému seznámení s programem bude sloužit průvodce, který se zobrazí při prvním spuštění a postupně uživatele provede celým procesem vložení nezbytných dat. Po dokončení průvodce bude potom hlavní obrazovka obsahovat již personalizované informace, tj. nový uživatel před sebou uvidí již známá data, od kterých se bude moci odrazit při postupném poznávání zbytku aplikace a vyhne se tím zbytečnému tápání – jednotlivé funkce bude objevovat v závislosti na postupně vyvstávající potřebě. Průvodce zároveň umožní předložit uživateli základní informace o použití aplikace, aniž by musel sám vyhledávat jakoukoli dokumentaci či nápovědu.

Intuitivnosti a strmé křivce učení pak přispěje také snaha vyhýbat se vrstveným nabídkám – všechny funkce by měly být dostupné co nejmenším počtem úkonů.

6.3 Volba technologie

6.3.1 Platforma

Důležitým a nejjednodušším rozhodnutím při volbě platformy, na které by měl program být postaven, byla volba mezi webovou aplikací poskytovanou uživateli ve formě služby, nebo nativní samostatné aplikace pro desktopové prostředí.

Trend posledních let (jak naznačuje i kapitola 4 Srovnání existujících produktů) směřuje k přesunu velké části aplikačního software pro osobní počítače do prostředí cloud computingu a webových služeb (17, s. 36). Webové aplikace mají proti klasickému software běžícímu přímo na systému uživatele velké množství výhod – mj. takřka neomezená kompatibilita a interoperabilita mezi různými platformami, nízké nároky na výpočetní výkon klientské stanice, bezproblémová průběžná údržba a aktualizace ze strany poskytovatele a v neposlední řadě dislokace dat mimo ICT prostředí uživatele, zbavující jej do velké míry potřeby data zálohovat a uchovávat. Zároveň konkrétně v případě školních informačních systémů výrazně zjednodušuje zpřístupnění informací rodičům a žákům. Webová služba navíc nevyžaduje žádnou formu instalace u klienta, což by obzvláště v případě aplikace, která si klade za cíl co nejsnazší zprovoznění a co nejjednodušší obsluhu jednoznačně znamenalo zásadní přínos.

Zároveň ale provoz webové aplikace znamená trvalé náklady pro provozovatele systému a její běh je zcela závislý na dlouhodobé údržbě a monitoringu. V případě zdarma dostupné aplikace vznikající v rámci jednorázového projektu, jakým je bakalářská práce, ale tyto skutečnosti přinášejí nezanedbatelné komplikace a provozní nároky, přičemž zároveň vystavují celý produkt riziku krátkého životního cyklu. Další komplikace pak představuje přebrání zodpovědnosti za uložená data jak z hlediska jejich ztráty či znehodnocení, tak v rámci zabezpečení úložiště a přenosu dat proti zneužití.

Protože mým cílem bylo, aby výstup práce měl naději představovat reálný přínos alespoň pro část málotřídních škol, a zároveň nemám jistotu, že bych byl v budoucnu schopen věnovat aplikaci dostatek času a úsilí, aby byla životaschopná jako webová služba, rozhodl jsem se zvolit cestu desktopové aplikace. Tak mohu očekávat, že bude výsledný software použitelný ještě poměrně dlouhou dobu po ukončení vývoje a podpory, obzvláště když jde o software, jehož funkcionalita je poměrně obecná a nerestriktivní a tudíž by neměla podléhat zastarávání v průběhu času.

6.3.2 Programovací jazyk

S přihlédnutím k rozsahu a předpokládaným nárokům na aplikaci není nutné příliš analyzovat výkonnostní efektivitu programu na úrovni programovacího jazyka. Hlavními kritérii jsou tedy kompatibilita s co největším portfoliem systémů na straně uživatelů a náročnost vývoje. Tyto požadavky, společně s předchozími zkušenostmi s vývojem aplikací v tomto jazyce, mě vedly k volbě Java Standard Edition, konkrétně v aktuálně nejnovější verzi Java SE 8.

Java je objektově orientovaný vysokoúrovňový programovací jazyk (18, s. 1), který byl vytvořen jako multiplatformní. Těto vlastnosti je docíleno díky tomu, že překladač (kompilátor) Javy nepřekládá zdrojový kód do strojového kódu, který by byl závislý na architektuře procesoru, operačním systémům apod., ale do tzv. bytecode. Tento na architektuře nezávislý bytecode je poté spouštěn pomocí Java Virtual Machine, tj. interpretu, který je předem instalován na cílovém zařízení. Snadné přenositelnosti programů je docíleno také díky tomu, že ve specifikaci není ponechán prostor pro aspekty závislé na konkrétní implementaci, tj. jsou zde například pevně dané velikosti datových typů (19, s. 4-5).

Díky těmto vlastnostem mohou být v Javě napsané programy provozovány na všech typech zařízení, pro které existuje distribuce Java Virtual Machine.

6.3.3 Databázové úložiště

Vzhledem k tomu, že se projekt profiluje jako databázová aplikace, bylo nezbytné zvolit také databázový engine, který by poskytl nástroje pro práci s ukládanými daty. Stejně jako v případě programovacího jazyka je zde důležitá spíše nenáročnost nasazení v aplikaci a také zprovoznění v prostředí uživatelů než výkon. V tomto ohledu je tradiční model klient-server, který vyžaduje zvlášť běžící a nakonfigurovanou službu databázového serveru a jeho průběžnou administraci, značně komplikovaný, přičemž by vzhledem k velikosti databáze v řádu několika set záznamů zároveň představoval neúměrnou zátěž zařízení ze strany samotné databázové infrastruktury. Jako mnohem vhodnější řešení se zde jeví použít embedded databázový engine, tedy takový, který je přímo součástí procesu aplikace (20, s. 1). Databáze ukládaná do souboru v konzistentním stavu navíc umožňuje velmi snadné zálohování dat prostřednictvím prostého zkopírování souboru „database.db“ na externí či vzdálené úložiště a umožňuje provoz aplikace rovnou z přenosného či síťového úložiště na různých zařízeních bez nutnosti instalace.

Na základě předchozích zkušeností jsem zvolil využití volně dostupného řešení SQLite, multiplatformní, přenosné knihovny poskytující databázový engine pracující na principu dotazovacího jazyka SQL (21). Omezení vyplývající z designu SQLite, jako je například možnost pouze jednoho souběžného zápisu do databáze, u tohoto typu aplikace nebudou představovat žádný problém, pokud na ně bude brán ohled při návrhu programu a jeho implementaci (20, s. 11).

6.3.4 Formáty pro export

Při volbě formátů, do kterých bude možné uložit výsledný rozvrh, jsem vycházel především ze tří kritérií, resp. tří variant uchovávání či zveřejnění rozvrhu hodin:

1. fyzická tištěná verze pro vyvěšení v prostorách školy a případnou archivaci
2. elektronická verze pro uložení v lokálním prostředí a archivaci
3. elektronická verze pro zveřejnění v rámci webové prezentace školy či obce

Vzhledem k tomu, že univerzálním a centrálním jazykem k vytváření webových stránek je HTML (22), byla volba formátu exportu pro web zcela přímočará. V principu by bylo možné využít jej i pro archivaci a tisk, nicméně tento postup by s sebou přinesl několik úskalí. Kritickou překážkou zde je fakt, že při tisku HTML dokumentů implicitně dochází k vynechání barev pozadí, což by značně omezilo možnosti rozlišení hodin. Navíc lze před zahájením samotného tisku jen stěží určit, jak bude vypadat výsledek.

Z těchto důvodů bylo třeba zvolit pro tisk a archivaci formát vhodnější, takový, který by dokázal zaručit konzistentní a předvídatelné zobrazení i tisk na všech zařízeních. Tyto požadavky dokáže splnit PDF (Portable Document Format), otevřený formát pro elektronické dokumenty vytvořený v 90. letech společností Adobe Systems a nyní spravovaný Mezinárodní organizací pro normalizaci v rámci standardu ISO 32000. (23)

7 Vývoj

7.1 Uživatelské rozhraní

7.1.1 Design GUI

Návrh uživatelského rozhraní reflektuje požadavky a záměry uvedené v kapitole 6.2 User experience, zásadní část proto představuje již zmíněný průvodce inicializací klíčových dat, který by měl uživatele již při prvním spuštění hladce uvést do ovládání programu a umožnit mu zadat všechny potřebné informace, aniž by jejich nastavení musel hledat v jakémkoli menu. V tomto průvodci bude možné zadat seznam tříd, učeben, učitelů a předmětů. Dodatečné úpravy a doplňování dat potom již zajišťují dialogová okna pro konkrétní číselníky v odpovídajících položkách menu.

Po dokončení prvotního průvodce, tj. ve chvíli, kdy jsou zadána všechna data potřebná pro vytváření rozvrhu, je již možné zobrazit hlavní obrazovku, jejímž centrálním prvkem je tabulka představující jednotlivé dny v týdnu, třídy a vyučovací hodiny a ve které již bude probíhat samotné rozvržení výuky. K sestavení rozvrhových lístků potom budou sloužit tři nástrojové panely, každý umístěný podél jedné strany rozvrhu – učitel, předmět a učebna. Na vrchní straně, pod hlavičkou okna, poté bude k dispozici panel menu obsahující tzv. drop down menu umožňující práci s daty, export rozvrhu apod.

Vytváření relací mezi jednotlivými entitami probíhá tak, že uživatel prvním klikem označí prvek, který chce přidat do rozvrhu, a druhým kliknutím jej potom umístí do konkrétní hodiny. Během vývoje byla druhou zvažovanou variantou metoda „drag and drop“, tj. přetahování prvků pomocí držení tlačítka myši, která by ale neumožnila tak snadno přidávat jeden prvek opakovaně do více polí v rozvrhu a tudíž by vyžadovala mnohem více pohybů myši k dosažení stejného výsledku. Každá hodina může kromě časového zařazení a třídy nést další tři informace: předmět, který je jako nejdůležitější informace uveden tučně na prvním řádku, učebnu, která je vypsána na druhém řádku běžným písmem, a učitele, který je pro lepší přehlednost a úsporu místa představován barevným rámečkem kolem buňky.

Přehledové sestavy umožňující získat informace o stavu naplnění úvazku učitele a učebních plánů jednotlivých tříd se nacházejí v panelu Sestavy. Volba požadované sestavy probíhá pomocí dvou rozbalovacích menu a samotná data jsou vypsána v tabulce, ze které je možné je snadno zkopírovat např. do tabulkového procesoru k dalšímu zpracování.

Rozvrh									
Soubor Data Nápvěda									
Rozvrh Souhrn									
Den	Třída	1	2	3	4	5	6		
ČJ	Pondělí	1	Čj - Tr. 2	M - Tr. 2	Tv - Tělocvična	Hv - Tr. 2			Tr. 1
M		2	Čj - Tr. 2	M - Tr. 2	Tv - Tělocvična	Hv - Tr. 2			Tr. 2
		3	Čj - Tr. 2	M - Tr. 2	Vv - Tr. 1	Př - Tr. 1			Tělocvična
Vv		4	M - Tr. 1	Aj - Tr. 1	Vv - Tr. 1	Př - Tr. 1			Družina
Př		5	M - Tr. 1	Aj - Tr. 1	Tv - Tělocvična	Př - Tr. 1			Dělny
Prv	Úterý	1							
Pv		2							
		3							
Tv		4							
Hv		5							
Aj	Středa	1							
		2							
		3							
		4							
		5							
	Čtvrtek	1							
		2							
		3							
		4							
		5							
	Pátek	1							
		2							
		3							
		4							
		5							

Jana Nováková

Marcela Vzdělaná

Josef Veselý

Rozvrh															
Úpravy Export															
Rozvrh		Sestavy													
Po	Anglický jazyk	1. První	Cj 100	M 100	Cj 100	Tv Tělocvična									101
	Hudební výchova	2. První	Cj 100	M 100	Cj 100	Tv Tělocvična									101
	Informatika	3. Druhá	Cj 101	M 101	Cj 101	Prv 101									Tělocvična Družina
	Matematika	4. Druhá	Cj 101	M 101	Cj 101	Aj 101	Vi 101								
	Pracovní vyučování	5. Druhá	Cj 101	M 101	Cj 101	Aj 101	Vi 101								
	Přirodověda	1. První	M 100	Cj 100	Prv 100	Pv 100				Hv 100					
	Tělocvik	2. První	M 100	Cj 100	Prv 100	Pv 100				Hv 100					
	Tělocvik - zdravotní výchova	3. Druhá	M 101	Cj 101	Prv 101	Tv Tělocvična				Hv 100	Cj 101				
	Vlastivěda	4. Druhá	M 101	Cj 101	Př 101	Tv Tělocvična				Hv 100					
St	Výtvarná výchova	5. Druhá	M 101	Cj 101	Př 101	Tv Tělocvična				Hv 100	Cj 101				
	Český jazyk														
	Školní družina	1. První	Tv-zv Tělocvična	Cj 100	Cj 100	ŠD Družina									
		2. První	Tv-zv Tělocvična	Cj 100	Cj 100	ŠD Družina									
		3. Druhá	Aj 101	Cj 101	M 101	Cj 101	Pv 101								
		4. Druhá	Aj 101	Cj 101	M 101	Vi 101	Pv 101								
		5. Druhá	Aj 101	Cj 101	M 101	Vi 101	Pv 101								
		1. První	M 100	Cj 100	ŠD Družina	Cj 100				Vv 100	Tv Tělocvična				
		2. První	M 100	Cj 100	ŠD Družina	Cj 100				Vv 100	Tv Tělocvična				
Václav Hořký [V4]				Petra Zelenská [P2]				Jaroslava Červíná [JC]							

Obr. 8 - Výsledné uživatelské rozhraní

Uživatelské rozhraní je realizováno prostřednictvím UX knihovny Swing, která je součástí distribuce Java. Tato knihovna má poskytovat nástroje k vytvoření uživatelského rozhraní natolik nezávislého na platformě, nakolik je to možné. (24) Výsledné rozhraní je

kombinací prvků navržených prostřednictvím NetBeans IDE GUI Builderu a programově generovaných částí.

Z programového hlediska představuje základní stavební kámen grafického rozhraní třída `Exec` (`extends javax.swing.JFrame`). Ta obsahuje nejen kompletní rozvržení hlavního okna aplikace, ale zajišťuje i značnou část aplikační funkcionality a je hlavní spustitelnou třídou aplikace.

Metody `printClassrooms`, `printTeachers` a `printSubjects` zde zajišťují vykreslení všech tří nástrojových panelů pro nasazování rozvrhu, metoda `printLessons` potom vypisuje rozvrhové lístky samotné. Zároveň se zde nachází většina obsluhy uživatelských akcí. Třída `Exec` zajišťuje také výpis souhrnných sestav. Další dílčí prvky GUI představuje pět tříd umístěných v balíku `guipackage`: `ClassRoomPane`, `LessonPane`, `SubjectPane` a `TeacherPane` jsou výchozími třídami pro tvorbu objektů představujících jednotlivé učebny, vyučovací hodiny, předměty a učitele, které za běhu aplikace nesou také klíče k odpovídajícím záznamům v databázi a části programové logiky. Pátá třída s názvem `ExportSettings` obsahuje nástroje pro konfiguraci exportu rozvrhu.

Rozložení prvků v okně programu zajišťují tzv. layout managery – sada tříd z knihovny AWT, které umožňují rozmístit objekty ve vztahu k ostatním a dynamicky jejich pozici přizpůsobovat velikosti okna, rozlišení obrazovky zařízení apod. Pro pozicování nástrojových panelů a posuvníků je použit layout manager `BorderLayout` zajišťující ukotvení všech prvků k hranám nadřazené entity:

```
1. 1. jPanel_rozvrh.setLayout(new BorderLayout(0, 0));
```

Ostatní interaktivní elementy jsou organizovány do mřížky pomocí `GridLayout`:

```
1. jPanel_subjects.setLayout(new GridLayout(20, 1, 0, 10));
```

Vzhledem k tomu, že rozvrh hodin je v principu trojrozměrným polem jednotlivých vyučovacích hodin (jedním rozměrem je den, druhým třída a třetím pořadí hodiny v rámci dne a denního rozvrhu dané třídy), je jeho vykreslení realizováno prostřednictvím tří vnořených cyklů „for“. První cyklus prochází jednotlivé dny a v každé iteraci vytvoří instanci třídy `JPanel` představující odpovídající den, následující cyklus vytváří pole pro jednotlivé třídy v každém dnu a poslední inicializuje objekt typu `LessonPane` pro každý rozvrhový lístek. Při vykreslení je navíc každé instanci třídy `LessonPane` přiřazena obsluha reagující na kliknutí levým tlačítkem myši. Je-li registrováno kliknutí, aplikace vyhodnotí, zda je označen také některý z objektů typu `SubjectPane`, `ClassroomPane` či `TeacherPane` a na základě tohoto vyhodnotí typ parametru, který je třeba předat jedné tří

přetížených metod třídy LessonAssigner (viz [7.3.4 LessonAssigner](#)). Pokud není označen žádný prvek k přidání do rozvrhu a uživatel zvolí z nabídky Úpravy možnost Odstranit hodinu, změní se kurzor myši na křížek a rozvrhový lístek je po kliknutí smazán.

```

1. for (int dayiterator = 0; dayiterator < days.length; dayiterator++) {
2.     [...]
3.     for (int classiterator = 0; classiterator < classes.size(); classiterator++)
4.     {
5.         [...]
6.         for (int orderiterator = 0; orderiterator < maxdailylessons; orderiterato
7.             r++) {
8.                 [...]
9.                 hodina.addMouseListener(new MouseAdapter() {
10.
11.                     @Override
12.                     public void mousePressed(MouseEvent me) {
13.                         try {
14.                             switch (selected[0]) {
15.                                 [...]
16.                             }
17.                             catch (Exception ex) {
18.                                 System.err.println("Chyba: " + ex.getMessage());
19.                             }
20.                         }
21.                     }
22.                 });
23.                 dayPanes[dayiterator].add(hodina);
24.                 hodina.validate();
25.             }
26.         }
27.     }
28.     jPanel_content.setAutoscrolls(true);
29. }
30. jPanel_rozvrh.repaint();
31. jPanel_rozvrh.validate();

```

Prvky, které nebylo nutné generovat algoritmicky, jsou vytvořeny pomocí editoru integrovaného ve vývojovém prostředí. Hlavní nástrojový panel je tvořen kombinací elementů typu JMenuBar, JMenu a JMenuItem, přepínání mezi zobrazením rozvrhu a souhrnných sestav zajišťuje JTabbedPane a panel sestav obsahuje dva JComboBoxy pro volbu typu sestavy a jTable pro její výpis.

Získávání dat pro přehledové sestavy probíhá s pomocí relativně komplexních databázových dotazů, přičemž na úrovni programu již jde pouze o převedení pole výsledků do tabulkového modelu a jeho vykreslení. Následující ukázka představuje SQL dotaz pro získání informací o úvazku učitele:

```

1. SELECT COUNT(DISTINCT hodina.poradi) AS pocet, den.nazev AS nazev FROM hodina JOIN
2. den ON hodina.den_idden = den.idden WHERE hodina.ucitel_iducitel =? GROUP BY ho
3. dina.den_idden;

```

7.1.3 Stavební prvky GUI

Jak je zmíněno výše, jednotlivé rozvrhové lístky stejně jako učitelé, třídy a vyučovací předměty jsou tvořeny objekty odpovídajících tříd. Tyto objekty neobsahují pouze viditelné

informace, ale každý z nich nese také primární klíč odpovídajícího záznamu v databázi a část kódu zajišťující interaktivitu celého rozhraní.

Třídy `ClassRoomPane`, `SubjectPane` a `TeacherPane` kromě zdrojového kódu specifikujícího jejich vzhled v GUI obsahují pouze svůj primární klíč, konstruktor nastavující požadované vlastnosti a metodu `toggleState`, která mění vzhled prvku v GUI podle toho, zda je aktuálně označen nebo ne. `LessonPane` je o něco komplikovanější, obsahuje totiž nejen svůj primární klíč (který je navíc složený z dne, třídy a pořadí hodiny, zatímco ostatní elementy mají své unikátní ID), ale také metody pro změnu údajů o dané hodině v databázi a její vykreslení s novými údaji.

7.1.4 Průvodce inicializací

Vyvolání průvodce inicializací se řídí dvěma kritérii. Prvním z nich je, zda vůbec existuje soubor `database.db`, druhým je záznam „init“ v tabulce „config“ v databázi. Pokud `database.db` neexistuje, dojde k zavolání metody `initDB` ze třídy `DBTools`, která zajistí vytvoření databázového souboru a inicializaci jednotlivých databázových tabulek. U relací, které mají být předvyplněné, tj. tabulky „den“ a „barva“, dojde i k vložení záznamů. V tabulce „config“ je v tomto kroku vyplněná hodnota záznamu „init“ na 0, pokud je průvodce úspěšně dokončen, přepíše se tato na 1. Při ukončení průvodce před splněním všech kroků, hodnota „init“ zůstane 0 a průvodce se znovu spustí při dalším zapnutí programu.

Průvodce inicializací je tvořen sedmi třídami umístěnými ve zvláštním java package „wizard“. Šest z nich (`Welcome`, `InsertClassRooms`, `InsertClasses`, `InsertSubjects`, `InsertTeachers` a `Finish`) představuje jednotlivé kroky průvodce a dědí od třídy `javax.swing.JPanel`, sedmá, nazvaná `InitWizard`, je potom univerzálním kontejnerem pro ně a jako taková dědí `javax.swing.JFrame`.

Funkcionalita průvodce je využita nejen při úvodní inicializaci databáze, ale jeho jednotlivé části slouží i pro dílčí úpravy při další práci. Diferenciace mezi těmito dvěma aplikacemi je realizována na úrovni třídy `InitWizard`, kde je volána buď metoda `public void firstRun`, která načte panel `Welcome` a ovládacímu prvku `JButton` nastaví obsluhu tak, aby přepínalo mezi jednotlivými kroky průvodce, nebo metoda `public void editRun`, která v závislosti na vstupní hodnotě zobrazí panel pro editaci požadovaného číselníku a ovládacímu tlačítku přiřadí událost zavření okna.

7.1.5 Podokno exportu

Podokno exportu umožňuje uživateli zvolit požadovaný vzhled a cíl exportu rozvrhu. Z hlediska grafického rozhraní je kompletně výsledkem práce v GUI Builderu a celá třída `ExportSettings`, na které stojí, nese jen minimum programové logiky.

7.2 Databáze

Pro databázové operace je použita knihovna `SQLite JDBC`, jejímž autorem je Taro L. Saito a která je šířena pod licencí `Apache License version 2.0`. (25)

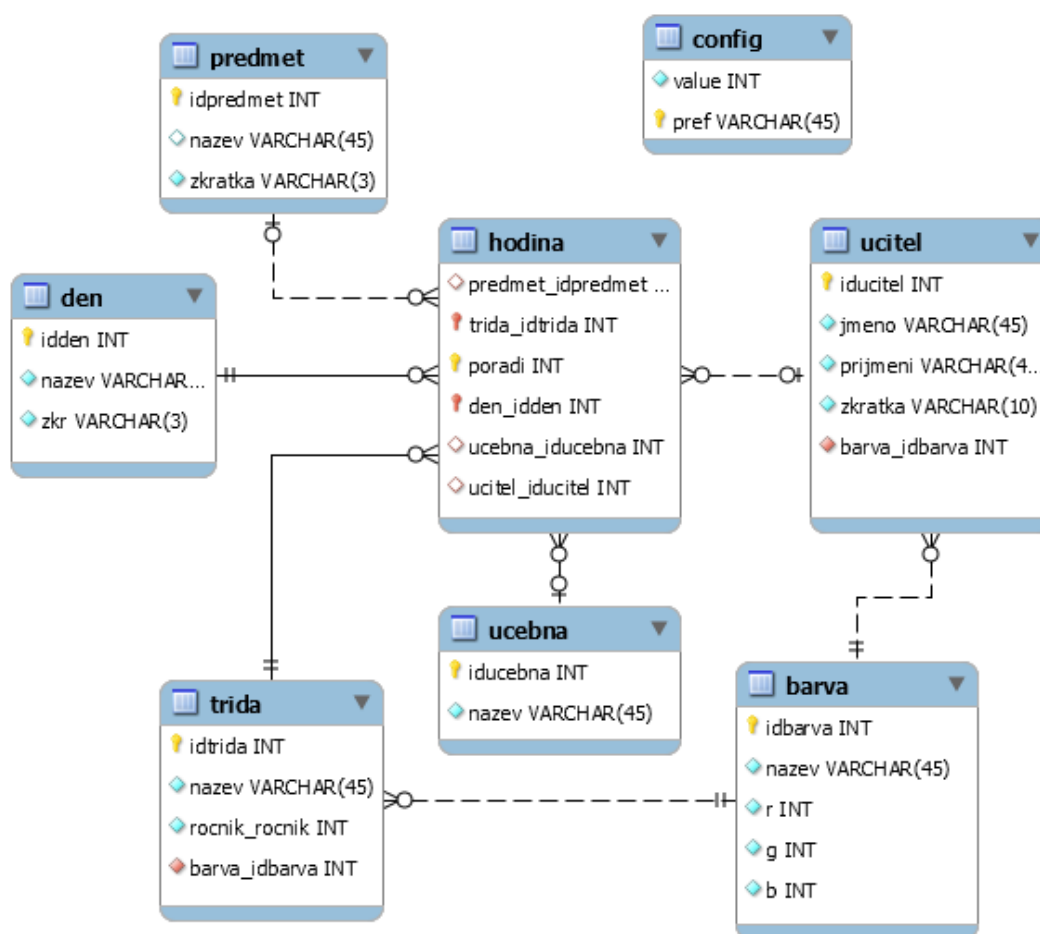
Samotná relační databáze obsahuje osm tabulek (relací), z nichž sedm je určeno přímo pro uložení dat souvisejících s rozvrhem a jedna je slouží k ukládání konfiguračních záznamů – ta v současné verzi aplikace obsahuje pouze jeden záznam určující, zda uživatel již prošel inicializačním průvodcem a má se při spuštění aplikace zobrazit rovnou hlavní obrazovka.

Centrálním prvkem je zde tabulka „hodina“, která představuje rozvrhový lístek jako takový a nese reference do dalších tabulek, jako jsou tabulky „ucitel“, „ucebna“ a další.

Vzhledem k postupnému principu, jakým probíhá přiřazování entit z ostatních tabulek do záznamů v tabulce hodina, jsou vlastnosti vztahů, zejména jejich volitelnost, nastaveny poměrně volně a nereflexují tak dokonale skutečnost. Zatímco reálně hodina, ve které není nastaven např. vyučující nebo předmět, víceméně postrádá smysl, v databázi tyto vztahy nejsou nastaveny jako mandatorní, aby mohlo docházet k průběžnému sestavování rozvrhu bez nutnosti mít předpřipravené rozvrhové lístky nebo změny před uložením do databáze udržovat v jiné datové struktuře. Tento přístup navíc koreluje s nerestriktivní filozofií celé aplikace a nechává prostor pro neočekávané situace, které by v tak specifickém a na improvizaci bohatém prostředí, jakým jsou málotřídní školy, mohly nastat.

Pro zabezpečení konzistence databáze je návrh normalizován do třetí normální formy, tj. všechny atributy v jednotlivých tabulkách jsou nedělitelné a zároveň závislé na celém klíči dané tabulky (26, s. 44-46).

Databáze je na disku uložena jako soubor „database.db“ ve složce s ostatními soubory aplikace.



Obr. 9 ER schéma databáze

7.3 Back end

Veškerá oddělená programová logika se nachází v balíku tříd `utilitypackage` a je rozdělena mezi čtyři třídy: práci s databází zajišťuje třída `DBTools`, zásadní část tvorby rozvrhu probíhá ve třídě `LessonAssigner`, algoritmy pro export dat obsahuje třída `TimetableExport` a jako klient pro nahrání souborů na FTP server slouží `FtpClass`.

7.3.1 DBTools

Třída `DBTools` je se svými více než 700 řádky zdrojového kódu a 45 metodami jednou z nejrozsáhlejších v projektu a je jedinou, která pracuje přímo s knihovnou `SQLite`. Slouží tak jako jakési rozhraní mezi zbytkem aplikace a databází a zajišťuje veškeré databázové operace v rámci aplikace. Její instance je inicializována hned po spuštění hlavní třídy a za běhu programu je předávána dalším objektům, které potřebují k databázi přistupovat. Jak již bylo zmíněno, všechny provedené datové změny se do databáze zapisují instantně v celém průběhu práce s rozvrhem, což sice má vzhledem k souborové povaze

SQLite databáze negativní vliv na výkon a odezvu aplikace, obzvlášť pokud se databázový soubor nachází na pomalém úložišti, ale zároveň je tím zaručena neustálá konzistence dat a jsou víceméně vyloučeny jakékoli ztráty neuložených změn.

Metody v této třídě je možné rozdělit na několik základních druhů. Ty, které začínají „add“ a slouží k přidání nové entity do databáze, ty začínající „get“ nebo „list“ a slouží k získání dat z databáze, resp. vypsání dostupných prvků, dále takové třídy, které začínají „delete“ či „clear“ a umožňují odstranění záznamů.

Specifické jsou metody `public void initDB`, která nese informace o kompletním databázovém schématu a zajišťuje prvotní vytvoření databázového souboru se všemi tabulkami, `public int isInitialized` sloužící po startu aplikace k rozhodnutí, zda spustit průvodce inicializací databáze, nebo rovnou hlavní okno programu. Metoda `public void setInitialized` zajišťuje přepnutí stavu inicializace v tabulce config a `public boolean doesTeacherCollide` slouží ke kontrole rozvrhových kolizí.

V následující ukázce zdrojového kódu je znázorněn způsob práce s databází s použitím knihovny SQLite JDBC:

```
1. public class DBTools {
2.
3.     public DBTools() throws ClassNotFoundException {
4.         Class.forName("org.sqlite.JDBC");
5.         //načtení sqlite JDBC interface
6.     }
7.
8.     public void updateClassRoom(int id, String name) throws SQLException {
9.         //metody využívající napojení na databázi musí zachycovat výjimky typu
        SQLException
10.        Connection conn = DriverManager.getConnection("jdbc:sqlite:database.db");
11.        //Napojení na soubor databáze database.db, cesta je relativně ke
        kořenovému adresáři programu
12.        PreparedStatement prep = conn.prepareStatement("UPDATE ucebna SET nazev=?
        WHERE iducebna=?");
13.        //vytvoření výrazu PreparedStatement
14.        prep.setString(1, name);
15.        prep.setInt(2, id);
16.        //doplnění parametrů do PreparedStatement
17.        prep.executeUpdate();
18.        //provedení změn v databázi
19.        prep.close();
20.        //uzavření PreparedStatement
21.        conn.close();
22.        //uzavření spojení s databází a jeho uvolnění pro další práci
23.
24.    }
25. }
```

V příkladu je rozebrána metoda `public void updateClassRoom` sloužící pro aktualizaci údajů o učebně. Pro dostatečný kontext je zahrnuta i nadřazená třída `DBTools` a její konstruktor, ve kterém probíhá inicializace rozhraní pro přístup k databázi. Důležitým

prvkem zde je demonstrace využití parametrizovaného `PreparedStatement` k zápisu do databáze. Tento postup by bylo možné nahradit dvouřádkovým

```
1. Statement stat = conn.createStatement();  
2. stat.executeUpdate("UPDATE ucebna SET nazev=" + name + " WHERE iducebna=" + id +  
    ";" );
```

Takový postup by ale nechal vstup databáze nechráněný před znaky, které mají v jazyce SQL svůj speciální význam, a umožnil tak vznik jevu známého jako SQL Injection. (27) Tento problém sice u desktopové aplikace, u které nelze předpokládat cílené útoky zvenčí, není tak palčivý, jako by byl u systému dostupném na webu, nicméně stále nelze vyloučit neúmyslné použití některého z potenciálně nebezpečných znaků ve vstupních polích programu a riziko narušení integrity databáze či ztráty dat. Použitím `PreparedStatement`, ve kterém jsou uživatelské vstupy vkládány jako parametry do předem kompilovaného dotazu, lze této situaci zamezit. Programátorsky přívětivější a čitelnější `Statement` je proto v aplikaci používán jen tam, kde jsou dotazy pevně dané a neobsahují žádný proměnlivý prvek.

7.3.2 FtpClass

`FtpClass` je třída sloužící k nahrání exportovaného rozvrhu na vzdálený server pomocí protokolu FTP. Umožňuje tak přímo z rozhraní aplikace zveřejnit rozvrh na webových stránkách školy. Rozvrh lze přenést ve formátu html, tedy přímo jako webovou stránku zobrazitelnou ve webovém prohlížeči, tak jako PDF ke stažení. Metody obsažené v této třídě jsou `public void connect`, `public Boolean upload` a `public void close`.

Metoda `connect` zajišťuje připojení k serveru na základě přihlašovacích údajů předaných ve formě vstupních parametrů, metoda `upload` provádí samotné nahrání souboru a zároveň ověřuje úspěch přenosu, jehož výsledek vrací jako návratovou hodnotu typu `Boolean`. Metoda `close` pouze odpojí klienta od serveru.

Celou práci s protokolem FTP zprostředkovává volně dostupná knihovna `edtFTPj` distribuovaná společností Enterprise Distributed Technologies pod licencí LGPL (Lesser General Public License). (28)

7.3.3 TimetableExport

Nástroje nezbytné k exportu rozvrhu a k obsluze třídy `FTPClass` obsahuje třída `TimetableExport`. Zvenčí jsou v této třídě dostupné dvě metody: `public static void`

`htmlExport`, která obsahuje funkcionalitu pro uložení rozvrhu do formátu HTML vhodného k zobrazení na webu či ve webovém prohlížeči, a `public static void pdfExport` určená k exportu pro tisk či archivaci ve formátu PDF. Pro metodu `pdfExport` je zde ještě několik podpůrných metod: `private static PdfPTable signatures` pro vykreslení volitelných polí pro podpisy třídního učitele a ředitele, a potom čtveřice velmi podobných metod, které mají všechny shodně návratový typ `PdfPCell` a představují jednotlivé druhy buněk ve výsledné tabulce – buňky pro nadpisy sloupců, pro jednotlivé dny, třídy, rozvrhové lístky.

Z algoritmického hlediska jsou si obě veřejné metody velmi podobné a obě se příliš neliší od metody `printLessons` ve třídě `Exec` (viz [7.1.2 Programové řešení hlavního okna](#)). Všechny tyto tři metody mají principiálně shodný účel, a tím je vykreslení tabulky s rozvrhem.

Metoda `htmlExport` k tomuto účelu používá relativně nízkourovňovou funkcionalitu třídy `BufferedWriter` ze základního balíčku `java.io`, kdy je výsledný soubor postupně zapisován na principu znakového výstupu, přičemž náležitosti HTML jako je hlavička a formátovací značky jsou doplněny přímo v rámci zdrojového kódu metody. Vstupními parametry zde jsou:

- `DBTools dbt` – instance třídy `DBTools`, která umožňuje práci s databází
- `short coltext` – proměnná určující vzhled exportovaných dat (viz kapitola [7.1.4 Průvodce inicializací](#))
- `File f` – cílový soubor pro export

Metoda `pdfExport` pracuje se stejnými vstupními daty, jen zde ještě přibývá binární proměnná `Boolean signfield`, podle níž se rozhoduje, zda ve výsledném dokumentu budou pole pro podpis třídního učitele/učitelky a ředitele/ředitelky školy pro zařazení rozvrhu do oficiálních dokumentů školy. Tato pole v HTML exportu nejsou dostupná, neboť by u formátu, který není určen pro tisk, postrádala smysl. Funkcionalitu pro export do PDF zajišťuje knihovna `iText` ve verzi 5.5.10. Práce s knihovnou `iText` je zcela odlišná od čistě algoritmického přístupu ke generování zdrojového kódu HTML souboru. Tvorba PDF zde probíhá na principu vytváření objektů a specifikace jejich vlastností podobně, jako při vytváření grafického rozhraní s použitím knihovny `Swing`. Základním prvkem je zde objekt typu `Document`, který obsahuje vlastnosti stránky jako je velikost, orientace (na šířku či na výšku), okraje a podobně. Tomuto objektu se pak přidávají další objekty, např. tabulka,

odstavec a podobně. Ukázka představuje metodu zajišťující vykreslení podpisových polí pod rozvrh hodin, přičemž je zde vidět práce s tabulkou (objekt typu PdfPTable), buňkou tabulky (PdfPCell), textovou frází (Phrase) a nastavením písma v rámci fráze. Menším textovým prvkem než je Phrase je potom Chunk, který umožňuje například docílit různého formátování textu v rámci jedné fráze.

```

1. private static PdfPTable signatures() throws Exception {
2.     PdfPTable signtable = new PdfPTable(2);
3.     PdfPCell signcell = new PdfPCell();
4.     signcell.setBorder(0);
5.     signcell.setHorizontalAlignment(Element.ALIGN_CENTER);
6.     signcell.setPhrase(new Phrase("_____\nTřídni učite
l/ka", new Font(BaseFont.createFont(BaseFont.TIMES_ROMAN, BaseFont.CP1250, true))
));
7.     signtable.addCell(signcell);
8.     signcell.setPhrase(new Phrase("_____\nŘeditel/ka š
koly", new Font(BaseFont.createFont(BaseFont.TIMES_ROMAN, BaseFont.CP1250, true))
));
9.     signtable.addCell(signcell);
10.    signtable.setWidthPercentage(100);
11.    signtable.setSpacingBefore(30f);
12.    return signtable;
13. }

```

7.3.4 LessonAssigner

Přestože třída LessonAssigner obsahuje zcela stěžejní funkcionalitu programu, jakou přiřazování učitelů, předmětů a učeben ke konkrétním rozvrhovým lístkům nepochybně je, ve skutečnosti obsahuje jen minimální množství zdrojového kódu a slouží v principu jen jako rozhraní mezi prvky GUI a databázovou třídou DBTools, přičemž jako doplňující funkci zajišťuje kontrolu kolizí v rozvrhu tak, jak je popsána již v kapitole [6.1 Funkcionalita](#). Tato kontrola probíhá na dvou místech a to při umístění učitele do hodiny a při přiřazení učebny rozvrhovému lístku.

Z programového hlediska sestává třída LessonAssigner ze tří stejně pojmenovaných metod public static void assign, které jsou přetěžovány na základě rozdílných vstupních parametrů. Společnými parametry všech tří jsou LessonPane lp, tedy instance konkrétního rozvrhového lístku, DBTools dbt pro komunikaci s databází. Přetížení vzniká u parametru TeacherPane tp pro přiřazení učitele, ClassroomPane cp pro učebnu a SubjectPane sp pro předmět. Kontrolu kolizí zajišťuje metoda doesTeacherCollide, viz ukázka kódu:

```

1. public static void assign(LessonPane lp, ClassroomPane cp, DBTools dbt) throws SQ
LException, Exception {
2.     if (!dbt.doesTeacherCollide(lp.getOrder(), lp.getDay(), cp.getId(), lp.getTri
da(),
dbt.getLessonTeacher(lp.getOrder(), lp.getDay(), lp.getTrida())) {
3.         lp.setClassRoom(cp.getId(), dbt);

```

```

4.         lp.repaint();
5.     } else {
6.         JOptionPane.showMessageDialog(null, "Vyučující tohoto rozvrhového lístku
           již má výuku v jiné učebně.", "Kolize učitele", JOptionPane.WARNING_MESSAGE);
7.     }
8. }

```

7.4 Průběh a postup vývoje

Prvním krokem při vývoji bylo vytvoření databáze a základních nástrojů pro práci s ní. Tak vznikla třída DBTools a prvních několik metod. Souběžně byla vytvořena i třída Exec, jako hlavní kontejner uživatelského rozhraní.

V dalších krocích vznikaly paralelně jednotlivé prvky GUI a odpovídající části obsluhy databáze, zpočátku pouze ve formě pasivních elementů vypisujících data vložená do databáze ručně. Vývoj nástrojů pro uživatelskou interakci byl zahájen až ve chvíli, kdy byl program schopný vykreslit předpřipravený rozvrh.

Po dokončení funkcionality nezbytné k přiřazování jednotlivých elementů do rozvrhových lístků byl implementován průvodce inicializací databáze, který zároveň poskytl rozhraní pro dodatečnou editaci jednotlivých číselníků. V tomto stavu již aplikace dokázala plnit svou primární funkci, tj. zadání uživatelských dat a vytvoření relací mezi učiteli, učebnami, třídami, předměty, dny a jednotlivými hodinami. K finální podobě této části aplikace zbývalo pouze přidat nástroje pro odstranění vyučovací hodiny a vyprázdnění databáze rozvrhových lístků pro nový rozvrh hodin např. při přechodu na další školní rok.

Následně byl zaveden panel sestav poskytující přehled o naplnění úvazků učitelů a učebních plánů tříd.

Poslední rozsáhlou fází vývoje byla implementace nástrojů pro export dat. Před prvním uživatelským testováním v reálném prostředí byl dokončen pouze export do formátu HTML, který umožnil výsledné soubory ad hoc upravit tak, aby bylo možné jejich zveřejnění a praktické použití. Export do PDF byl doplněn až posléze tak, aby v něm mohly být reflektovány dílčí požadavky nezachycené v prvotních analýzách.

7.5 Testování aplikace

Kromě průběžných, značně nestrukturovaných a nemoderovaných, konzultací jak s několika učiteli, tak se zcela nezainteresovanými uživateli, které byly cíleny především na ověření přístupnosti uživatelského rozhraní, proběhl praktický test aplikace v produkčním prostředí. Testovacím subjektem byl učitelský sbor stejné školy, ze které pocházel rozvrh

hodin, který sloužil jako referenční při analýze stávajících komerčních produktů. Test sestával z kompletní přípravy rozvrhu hodin pro následující školní rok.

Test se odehrál před začátkem školního roku 2016/2017 a jeho výsledky jsou rozebrány v následujících podkapitolách.

7.5.1 Uživatelské rozhraní

V zájmu zvýšení náročnosti testu tak, aby byla zajištěna co největší přístupnost finální verze pro uživatele, byly vynechány instruktážní texty v průvodci inicializací databáze. Testovací uživatel měl tudíž k dispozici skutečně jen čisté uživatelské rozhraní aplikace bez jakýchkoli dodatečných informací týkajících se principu zacházení s programem.

Uživatel byl schopen se s minimální nápovědou (např. osvětlení principu přiřazování prvků do rozvrhu postupným klikáním, které je ve finální verzi popsáno v závěrečné části průvodce) v aplikaci zorientovat a sestavit rozvrh podle svých představ. Finální verze obsahující instruktážní texty by tudíž měla poskytnout novým uživatelům dostatečné podklady ke zcela samostatnému zvládnutí zacházení s aplikací.

7.5.2 Rozvržení výuky

Z hlediska funkcí pro samotné rozvržení výuky se aplikace jako taková jeví zcela dostačující a nevyvstaly žádné speciální požadavky. Pozitivní reakci vyvolal panel sestav umožňující rychlou kontrolu správnosti rozvrhu porovnáním se školním vzdělávacím programem.

7.5.3 Export dat

Funkcionalita exportu dat se ukázala jako nedostatečná v následujících ohledech:

- chybělo označení pořadí vyučovacích hodin
- nebyla zavedena možnost barevného odlišení jednotlivých tříd (resp. spojů)
- chyběl nadpis „Rozvrh hodin“
- na tištěné verzi chyběla pole pro podpis třídního učitele a ředitele školy, která jsou pro oficiální výtisky rozvrhu žádoucí

Tyto problémy byly reflektovány v následujícím vývoji a v aplikaci tak, jak je součástí této práce, jsou již zmíněné funkce doplněny. Některé další funkce bohužel nebylo možné implementovat kvůli překážkám v návrhu základních součástí programu a časové

náročnosti. Zejména jde o možnost zanést do rozvrhu začátky a konce vyučovacích hodin a přestávek, která by vyžadovala výrazný zásah do databázového modelu a zároveň přepracování základních algoritmů aplikace. Druhou takovou funkcí je export rozvrhu jen pro vybrané třídy či vyučující, který by značně zkomplikoval dialog konfigurace exportu i jeho vnitřní logiku.

7.5.4 Kompatibilita

Vzhledem k tomu, že je aplikace kompletně postavena na technologiích, které se prezentují jako multiplatformní a přenositelné, jeví se jako vhodné otestovat chování aplikace i na různých prostředích a operačních systémech. Vzhledem k tomu, že nejrozšířenějšími operačními systémy v době dokončení vývoje jsou Microsoft Windows 7 a Microsoft Windows 10 (29), proběhlo testování na několika různých zařízeních s těmito dvěma operačními systémy. Na všech testovaných počítačích fungovala aplikace bez problémů.

Pro zjištění, zda aplikace dokáže fungovat i v open-source prostředí proběhly testy kompatibility i na počítačích s operačním systémem Linux. Zde byly vybrány dvě distribuce, především s přihlédnutím k dostupnosti, a to Ubuntu ve verzi 16.04 a Debian verze 8.6. I na těchto systémech se aplikace chovala korektně a je bez problémů použitelná.

8 Závěr

Práce přináší analýzu prostředí málotřídních škol v kontextu využití školních informačních systémů a především aplikací pro elektronickou tvorbu rozvrhů hodin. Tato analýza zcela jednoznačně ukázala, že zastoupení specializovaných aplikací při přípravě rozvrhů v těchto vzdělávacích zařízeních je prakticky neexistující a tudíž existuje potenciál pro novou aplikaci k tomuto účelu určenou. Skutečnost, že žádná z více než 43 škol, které se výzkumu zúčastnily, k přípravě rozvrhu žádný specializovaný nástroj nevyužívá, ale zároveň více než tři čtvrtiny z nich mají zájem takovou aplikaci vyzkoušet, naznačuje potenciální přesah práce do praktického využití v prostředí škol.

Přestože je tradice málotřídních škol v prostředí České Republiky silná a jejich četnost, která představuje více než třetinu základních škol u nás, srovnání dostupných školních informačních systémů a rozvrhových aplikací ukázalo, že jsou tyto školy silně opomíjeny i z hlediska výrobců školních informačních systémů, které specifické potřeby neúplných škol, jako je spojování výuky, vůbec nereflexují.

Výsledkem práce je jednoúčelová databázová aplikace úzce zaměřená na naplnění požadavků malých škol, která je jeví být ve svém oboru zcela unikátní a jejímž cílem je nejen usnadnit práci ředitelům málotřídních škol, ale také přivést pozornost k tématu. Přestože se ukázalo, že existuje jistý potenciál k vylepšení aplikace a doplnění některých funkcí, základní myšlenka se osvědčila jako správná a v praxi použitelná.

Náhled do prostředí managementu málotřídních škol získaný jak z výzkumné části, tak z následné spolupráce s učitelským sborem jedné školy při testování aplikace představuje silnou motivaci nenechat výslednou aplikaci stagnovat, ale pokračovat v jejím vývoji i po dokončení práce. K tomu přispívá také skutečnost, že jsem byl ještě několik měsíců po rozeslání dotazníku do škol kontaktován jednou z respondentek s dotazem, kde je možné aplikaci získat.

9 Přílohy

9.1 Příloha č. 1: Dotazník

1. Jaký program pro zpracování informací ve škole používáte?

Např. pro vysvědčení, evidenci žáků apod.

- Škola online
- Bakaláři
- SAS
- iŠkola
- Jiný

2. Které z následujících prostředků specifických pro malotřídky používáte ve svých rozvrzích?

- Spojování různých předmětů v rámci jedné vyučovací hodiny
- Spojování předmětů s rozdílnou hodinovou dotací mezi spojenými ročníky

3. Používá Vaše škola specializovaný počítačový program pro sestavování rozvrhu?

např. aSc rozvrhy, rozvrhový modul Bakalářů

- Ano - přeskočit na otázku 6.
- Ne – přeskočit na otázku 4.

4. Označte, prosím, důvody, které vás odrazují

- Příliš vysoká cena v poměru s reálným přínosem takové aplikace
- Složitě ovládání nabízených produktů
- Nedostatek informací o nabídce aplikací tohoto druhu
- Jiný

5. Máte zájem o vyzkoušení volně dostupné aplikace pro tvorbu rozvrhů zaměřené přímo na malotřídní školy?

Výstupem práce bude aplikace reflektující specifické požadavky malotřídek související se slučováním tříd. Zároveň u ní bude kladen vysoký důraz na jednoduché ovládání a vynechání nepotřebných funkcí snižujících přehlednost.

- Ano – zobrazit pole pro vložení e-mailové adresy.
- Ne – konec dotazníku

Děkuji za čas, který jste věnovali vyplnění dotazníku, a doufám, že svou prací přispějete k zohlednění potřeb malotřídních škol při vývoji školních informačních systémů.

6. Jaký nástroj používáte?

- aSc rozvrhy
- Rozvrhový modul Bakalářů
- Škola online
- SAS
- iŠkola
- Jiný

7. Máte přesto zájem o vyzkoušení volně dostupné aplikace pro tvorbu rozvrhů zaměřené přímo na malotřídní školy?

Výstupem práce bude aplikace reflektující specifické požadavky malotřídek, souvisejícím se slučováním tříd. Zároveň u ní bude kladen vysoký důraz na jednoduché ovládání a vynechání nepotřebných funkcí snižujících přehlednost.

- Ano – zobrazit pole pro vložení e-mailové adresy.
- Ne – konec dotazníku.

Děkuji za čas, který jste věnovali vyplnění dotazníku, a doufám, že svou prací přispějí k zohlednění potřeb malotřídních škol při vývoji školních informačních systémů.

10 Seznam použitých informačních zdrojů

- (1) ZWASS, Vladimír. Information System. *Britannica.com* [online]. b.r. [cit. 2015]. Dostupné z: <http://global.britannica.com/topic/information-system>
- (2) INGRAM, David. What Is a Management Information System?. *Chron.com* [online]. b.r. [cit. 2015]. Dostupné z: <http://smallbusiness.chron.com/management-information-system-2104.html>
- (3) BASL, Josef. *Informační systémy škol - informační systémy pro řízení činnosti základních a středních škol*. Praha: Ústav informačních studií a knihovnictví FF UK, 2006.
- (4) NEUMAJER, Ondřej. Školní informační systémy. *Metodický portál RVP.CZ* [online]. 2010 [cit. 2015]. Dostupné z: <http://clanky.rvp.cz/clanek/c/Z/8019/skolni-informacni-systemy.html/>
- (5) EMMEROVÁ, Kateřina. Malotřídky v současném prostředí českého venkova. *Studia paedagogica* [online]. 2013, **2013**(47), 81-96 [cit. 2016-03-28]. ISSN 1803-7437. Dostupné z: <http://www.phil.muni.cz/journals/index.php/studia-paedagogica/article/view/351>
- (6) Zákon č. 561/2004 Sb., o předškolním, základním, středním, vyšším odborném a jiném vzdělávání (školský zákon). In: *Sbírka zákonů*. b.r., ročník 2008, číslo 317.
- (7) Vyhláška o základním vzdělávání a některých náležitostech plnění povinné školní docházky. In: *Sbírka zákonů*. 2005, ročník 2005, číslo 48.
- (8) TRNKOVÁ, Kateřina. *Podmínky vzdělávání na malotřídních školách* [online]. České Budějovice, 2007, 11 s. [cit. 2016-03-28]. Dostupné z: http://www.phil.muni.cz/wupv/home/veda/podminky_capv.doc. Příspěvek prezentovaný na konferenci ČAPV. Jihočeská univerzita.
- (9) *Dlouhodobý záměr vzdělávání a rozvoje vzdělávací soustavy ve Středočeském kraji* [online]. Středočeský kraj, 2006 [cit. 2016-12-08]. Dostupné z: <https://www.kr-stredocesky.cz/documents/20875/63499/Dlouhodob%C3%BD%20z%C3%A1m%C4%9Br+2016.zip?version=1.0>

- (10) *Rejstřík škol a školských zařízení* [online]. MŠMT, b.r. [cit. 2016-03-28]. Dostupné z: <http://rejskol.msmt.cz/>
- (11) TRNKOVÁ, Kateřina, Dana KNOTOVÁ a Lucie CHALOUPKOVÁ. *Málotřídní školy v České republice: malotřídky*. 1. vyd. Brno: Paido, 2010, 197 s. ISBN 9788073152048.
- (12) JORDAN, Ron a Tony WHITELEY. Teaching Is a Full Time Job and so is Managing: Dilemmas and Delights of Managing a Small School. *Education-line* [online]. University of York: University of York, 1997 [cit. 2016-03-30]. Dostupné z: <http://www.leeds.ac.uk/educol/documents/000000507.htm>
- (13) Ceník. *Iškola.cz* [online]. b.r. [cit. 2016-04-10]. Dostupné z: <https://www.iskola.cz/iskola/cenik>
- (14) *ADMINISTRACE: systémová příručka pro správu jednotlivých objektů v aplikaci iskola.cz* [online]. b.r. [cit. 2016-04-04]. Dostupné z: https://www.iskola.cz/iskola/obrazky/clanky/02_administrace_objektu.pdf
- (15) Ceník. *Škola OnLine* [online]. b.r. [cit. 2016-04-10]. Dostupné z: <http://www.skolaonline.cz/%C5%98editel/Cen%C3%ADk.aspx>
- (16) GIDDENS, Anthony. *Sociology*. Cambridge UK: Polity, 2009. ISBN 9780745643571.
- (17) TAIVALSAARI, Antero a Tommi MIKKONEN. From Apps to Liquid Multi-Device Software. *Procedia Computer Science* [online]. 2015, **56**, 34-40 [cit. 2016-04-10]. DOI: 10.1016/j.procs.2015.07.179. ISSN 18770509. Dostupné z: <http://linkinghub.elsevier.com/retrieve/pii/S1877050915016609>
- (18) GOSLING, James, Bill JOY, Guy STEELE, Gilad BRACHA a Alex BUCKLEY. *The Java ® Language Specification: Java SE 8 Edition* [online]. Oracle America, Inc., 2015 [cit. 2016-04-10]. Dostupné z: <https://docs.oracle.com/javase/specs/jls/se8/jls8.pdf>
- (19) FLANAGAN, David. *Java in a nutshell: a desktop quick reference*. 2nd ed. Sebastopol, CA: O'Reilly, 1997. ISBN 15-659-2262-X.
- (20) GRANT ALLEN AND MIKE OWENS., . *The definitive guide to SQLite: [take control of this compact but powerful tool to embed sophisticated SQL databases*

within your applications!]. 2nd. ed. New York: Apress, 2010. ISBN 978-143-0232-254.

- (21) About SQLite. *SQLite* [online]. b.r. [cit. 2016-04-10]. Dostupné z: <https://www.sqlite.org/about.html>
- (22) HTML 5.1: 1. Introduction. *World Wide Web Consortium (W3C)* [online]. World Wide Web Consortium, 2016 [cit. 2016-12-07]. Dostupné z: <https://www.w3.org/TR/html51/introduction.html#introduction>
- (23) What is PDF? Adobe Portable Document Format. *Adobe Document Cloud* [online]. b.r. [cit. 2016-12-07]. Dostupné z: <https://acrobat.adobe.com/us/en/why-adobe/about-adobe-pdf.html>
- (24) Javax.swing (Java Platform SE 8). *Java Platform, Standard Edition 8* [online]. b.r. [cit. 2016-07-14]. Dostupné z: <https://docs.oracle.com/javase/8/docs/api/javax/swing/package-summary.html#package.description>
- (25) Xerial / sqlite-jdbc - Bitbucket. *Bitbucket* [online]. b.r. [cit. 2016-07-14]. Dostupné z: <https://bitbucket.org/xerial/sqlite-jdbc>
- (26) LACKO, Luboslav. *SQL: hotová řešení*. Vyd. 1. Brno: Computer Press, 2003. K okamžitému použití (Computer Press). ISBN 80-722-6975-5.
- (27) BOYD, Stephen a Angelos KEROMYTIS *SQLrand: Preventing SQL Injection Attacks* [online]. Department of Computer Science Columbia University, b.r. [cit. 2016-12-05]. Dostupné z: <http://web1.cs.columbia.edu/~angelos/Papers/sqlrand.pdf>
- (28) Free Java FTP library. *Enterprise Distributed Technologies* [online]. b.r. [cit. 2016-12-05]. Dostupné z: <http://enterprisedt.com/products/edtftpj/>
- (29) Operating System Market Share. *Market share for mobile, browsers, operating systems and search engines* [online]. 2016 [cit. 2016-12-08]. Dostupné z: <https://www.netmarketshare.com/operating-system-market-share.aspx?qprid=10&qpcustomd=0>

11 Seznam příloh

1. Plné znění dotazníku
2. Disk s vypracovanou aplikací, zdrojovými kódy a kompletními výsledky dotazníkového šetření ve formátu xls